Ariba Procurement Solution Integration Guide for SAP ERP
Ariba Procure-to-Order
Ariba Procure-to-Pay
# Content

**SAP ERP integration - overview.** .......................................................... 11
About SAP ERP integration. ................................................................. 11
Master data. ......................................................................................... 11
Transactional data. .............................................................................. 12
Data integration methods. .................................................................. 13
  - File-based integration. ................................................................. 13
  - Web services based integration. ................................................. 15
  - Direct connectivity integration. ................................................. 16
  - Mediated connectivity integration. ............................................ 16
Supported document transactions. .................................................... 17
Supported versions and integration landscapes. .................................. 18
Installation. ......................................................................................... 18

**Configuring SAP Process Integration.** ............................................. 20
Configuring the System Landscape Directory. .................................... 20
  - How to import product and component definitions for the Ariba Procurement Solution. ........................................... 20
  - How to configure the Technical Landscape for the Ariba Procurement Solution. .................................................. 21
  - How to configure the Business Landscape for the Ariba Procurement Solution. .................................................. 22
  - How to configure the Technical Landscape for SAP ERP. .............................................................................. 22
  - How to configure the Business Landscape for SAP ERP. .............................................................................. 23
Configuring transaction processes. .................................................... 24
  - About importing the design package. ........................................ 24
  - How to configure SAP ERP connectivity. .................................. 25
  - How to assign Business Systems in the Integration Directory. .............................................................................. 26
  - How to create Communication Channels for SAP ERP. .............................................................................. 26
Configuring SAP Process Integration value mapping. ....................... 27
  - Configuring global variables. .................................................... 28
About configuring Integration Scenarios in the Integration Directory. ..... 28
About Configuring the Integration Scenario. ....................................... 28
  - How to transfer the Integration Scenario. ................................ 30
  - Configuring the Integration Scenario. ...................................... 31
  - Activating the Integration Scenario. ......................................... 40
Integration Scenario parameter values. ............................................. 41
Configuring SAP Process Integration for Web services integration. .......... 45
  - How to generate keys for signing SAP Process Integration messages. ................................................................. 45
  - How to validate messages using the Ariba Procurement Solution. ........................................................................... 46
Configuration for inbound messages received by SAP Process Integration. .......... 46
# Configuring parameters

## About /ARBA/TVARV

### How to edit /ARBA/TVARV parameters

## Parameters description

### Advance payment parameters

### Allow duplicates for purchase orders

### Allow duplicates for invoice

### Attachments failure email-invoices

### Attachments failure email-purchase orders and change purchase orders

### Budget check requisition variables

### Currency conversion rates

### Days past

### DMS attachments storage

### Document type for creating expense report invoices

### Document type for creating OK-to-Pay invoices

### Document type for creating purchase orders

### Internal order types

### Language for texts

### Non-PO (FI) invoices

### Purchase order comments

### Rounding of gross amount for invoices

### Taxes and charges

### Using general ledger balance sheet accounts

### Vendor export task

### Maintaining a temporary directory in /ARBA/TVARV

#### How to maintain a temporary directory while running master data directly

#### How to maintain a temporary directory while running transaction data directly

# Configuring SAP ERP

## About configuring time-out intervals

### How to set the time-out interval

## Verifying data

### About the bank selling rate

### Blank descriptions

### Vendor restrictions

### Vendor ERS tax status

### Catalog item validation

### How to deactivate the company code validation

## Configuring service sheets

### How to create an output type for the entry sheet application
How to maintain condition records for outbound transactions: service sheets .............................................. 75

**Configuring Ariba Procurement Solution.** ................................................................. 77
Verifying connection information. .................................................................................. 77
Receiving tolerances. ....................................................................................................... 77
Receiving system parameters. .................................................................................. 78
How to enable generation of an ERP order by default. ............................................. 78
Specifying the preferred ordering method. ..................................................................... 79
Specifying a unique number for purchase orders. .................................................. 79
Enabling cancel order integration. ............................................................................. 79
How to enable creation of split orders based on currency. ........................................... 80
  About ERP orders. ........................................................................................................ 80
  Support for split amount charges. ............................................................................ 81
  Requirements for split by amount accounting. .......................................................... 81
  Customizing the goods receipt transaction to support accounting information. ........ 82
Specifying ad-hoc ship-to addresses to requisition and purchase orders. .................. 82
  Ship-to address fields with ad-hoc ship-to information. ........................................... 83

**Integrating master data.** ............................................................................................ 85
About master data. ........................................................................................................... 85
Importing master data. .................................................................................................... 86
  Required data. ............................................................................................................. 87
  Requirements for importing master data. ................................................................ 88
  Limitations for importing master data. ..................................................................... 88
  Backward compatibility. .............................................................................................. 88
  Simplified master data integration. ........................................................................... 89
  Defining authorizations for running Ariba Procure-to-Pay transactions. ............... 90
  Importing supplier data. .............................................................................................. 91
  Importing payment terms data. .................................................................................. 97
  Importing user data and user group data. ................................................................. 97
  Support integration for outline agreements in SAP ERP. ........................................ 98
Importing master data to Federated Process Control-enabled systems. ..................... 102
  Prerequisites for importing master data to Federated Process Control-enabled systems. 104
  Limitations for importing master data to Federated Process Control-enabled systems. 104
  User master data elements exported to a parent site ............................................... 105
  Maintaining table entries. ......................................................................................... 105
  How to import master data in Federated Process Control-enabled environments. .... 110
  Configuring Direct Connectivity. .............................................................................. 112
  How to import the Ariba Components. ..................................................................... 112
Incremental load events. .............................................................................................. 113
  Asset incremental load. .............................................................................................. 114
  Cost center incremental load. ..................................................................................... 116
General ledger incremental load .................................................. 118
Internal order incremental load .................................................. 121
WBS element incremental load .................................................. 122
User and user group incremental load ....................................... 122
Supplier incremental load ....................................................... 124
Company code import RFC example ......................................... 129
Customizing RFC ................................................................. 130
About extension structures ....................................................... 131
Adding fields to extension structures ........................................ 132
Typical extension structure ...................................................... 133
Filtering data .......................................................... 133
Static filtering ................................................................. 133
ABAP filtering ................................................................. 138
Post-hook filtering ............................................................. 140
Master data integration events ................................................ 142
Understanding the master data import ..................................... 142
Integrating master data using the user interface ....................... 142
How to run the master data import event ................................. 145
Checking output CSV files .................................................... 147
About importing data in multiple languages ............................. 149
Customizing Funds Management extracts ............................... 150
Integrating master data using the direct connectivity integration method from SAP ERP .................................................. 152
Limitations when integrating master data directly from SAP ERP .................................................. 153
Importing master data using the direct connectivity integration method .................................................. 153
Integrating master data using the mediated connectivity integration method .................................................. 161
Configuring mediated connectivity ......................................... 162

Integrating transactional data .................................................. 171
Integrating transactional data using the file channel ................... 171
 Transactional data integration event process flow ..................... 171
Advanced pricing details ....................................................... 177
Exporting purchase orders ..................................................... 178
Exporting change purchase orders ......................................... 180
Exporting close purchase orders ............................................. 183
Exporting cancel purchase orders .......................................... 185
Importing purchase order header status .................................. 186
Exporting receipts ............................................................ 187
Exporting invoices ............................................................. 188
Exporting expense reports .................................................... 193
Accommodating custom fields ................................................. 194
Calling the SAP BAPI .......................................................... 195
Scheduling ................................................................. 196
Importing remittance advice .................................................... 196
Integrating transactional data using the web services channel .......... 200
About transaction data ......................................................... 200
Components of an integration event ........................................ 203
Integrating purchase orders .................................................. 207
Integrating amount-based purchase orders ................................ 208
Integrating changed and canceled purchase orders .................... 209
Integrating closed purchase orders ......................................... 210
Integrating receipts ............................................................ 212
Integrating payments .......................................................... 218
Integrating remittance .......................................................... 219
Integrating real-time budget check .......................................... 221
Integrating taxes and charges ............................................... 227
Integrating advance payments .............................................. 230
Integrating non-PO (FI) invoices ........................................... 232
Integrating service procurement ........................................... 234
Integrating attachments support ............................................ 237
How to customize an existing integration event .......................... 242
Integrating transactional data directly using the user interface ....... 243
About integrating transactional data directly using the user interface 243
Configuring transactional data directly using the user interface ....... 244
Integrating transactional data using the mediated connectivity integration method .................................. 250
About integrating transactional data using the mediated connectivity integration method 250
Configuring transactional data using the mediated connectivity integration method 251

**Error Handling and Troubleshooting** ..................................... 258
About Error Handling and Troubleshooting ............................... 258
Application Errors ............................................................. 258
Application Errors When Importing Master Data Using Direct Connectivity .................. 259
Application Errors When Exporting Transactional data Using the File Channel ............... 259
Application Errors When Exporting Transactional Data Using the Web Service Channel .... 260
System Errors ................................................................. 261
System Errors When Importing Master Data Using Direct Connectivity .................. 261
System Errors When Exporting Transactional Data Using Direct Connectivity ............... 262
System Errors When Exporting Transactional Data Using the File Channel ....................... 263
System Errors When Exporting Transactional Data Using the Web Service Channel ............ 263
Viewing the Application Log ................................................ 264
Troubleshooting .............................................................. 265
Troubleshooting integration of service procurement in the Ariba Procurement Solution integrated with SAP ERP ........................................ 266
Enabling Debugging ......................................................... 266
Debugging Using the SAP User Interface .................................. 267
Debugging Parameters in Receipts ................................................... 267

**Account Assignment Categories.** .................................................. 268
About Account Assignment Categories ............................................. 268
Account Assignment Category Imports .......................................... 268
Configuring Account Assignment Categories .................................... 269
Understanding Ariba Procurement Solutions Accounting Types .......... 269
Configuring SAP Accounting Fields ................................................. 270
Understanding Field Status Group Strings .................................... 271
Viewing Field Status Group Strings ................................................. 272
Importing New Accounting Objects From SAP .................................. 273

**Integration Events Reference.** ..................................................... 275
Common Integration Events ............................................................. 275
Integration Events ............................................................................. 275

**Ariba SAP Objects.** ..................................................................... 280
Naming Conventions for Ariba SAP Objects ...................................... 280
Summary of Ariba SAP packages ..................................................... 281
/ARBA/MASTER_DATA_EXPORT Package ......................................... 281
  Function Groups and Modules ......................................................... 282
  Tables ......................................................................................... 284
  Structures ................................................................................... 284
  Proxy Structures .......................................................................... 286
  Table Types ................................................................................. 287
  Data Elements .............................................................................. 288
  Message Class .............................................................................. 289
  BAdI ............................................................................................. 289
/ARBA/PURCHREQP2P Package ....................................................... 289
/ARBA/PURCHREQP2P Tables .......................................................... 289
/ARBA/PURCHREQP2P Table Types ................................................. 290
/ARBA/PURCHREQP2P Structures .................................................. 290
/ARBA/PURCHREQP2P Domain ....................................................... 290
/ARBA/PURCHREQP2P Function Group .......................................... 290
/ARBA/PURCHREQP2P Function Modules ....................................... 291
/ARBA/PURCHREQP2P BAdI ............................................................ 291
/ARBA/INVOICE_CREATE Package ................................................ 291
/ARBA/INVOICE_CREATE Tables .................................................... 291
/ARBA/INVOICE_CREATE Structures ............................................. 291
/ARBA/INVOICE_CREATE Table Types ............................................ 292
/ARBA/INVOICE_CREATE Programs ............................................... 292
BAdI ............................................................................................ 293
How to download the mapping workbooks. ............................................ 321

Revision History. .................................................................................. 322
About SAP ERP integration

The Ariba Procurement Solution integrated with SAP ERP allows administrators to import master data and export transactional data from SAP ERP to the Ariba Procurement Solution seamlessly.

The Ariba Integration Toolkit is a Java-based tool that Ariba provides to upload master data or download transactional data. This tool reads the CSV files in SAP ERP, zips them, and sends them as MIME messages using the HTTP post to the Ariba Procurement Solution. For more information, see the Ariba On Demand Solutions Integration Toolkit Guide.

Buyer administrators can enable the integration of data through the following methods:

- File-based integration
- Web services
- Direct connectivity
- Mediated connectivity

Related Information

Data integration methods [page 13]

Master data

Master data consists of general information stored in SAP ERP that is used in the Ariba Procurement Solution to create business documents such as requisitions, receipts, or invoices. Master data is extracted from SAP ERP and sent to the Ariba Procurement Solution. The Ariba Procurement Solution integrated with SAP ERP provides the integration events for standard data imports from SAP ERP.
Master data is extracted from SAP ERP and sent to the Ariba Procurement Solution. The Ariba Procurement Solution provides the integration events for standard data imports from SAP ERP.

To import master data, you require the latest version of the SAP transports installed in SAP ERP. The SAP transports contain the Ariba-specific ABAP programs required for master data export extractions. When you export master data from SAP ERP, the ABAP programs runs the scheduled task in SAP ERP. To run master data, you must schedule the master data extraction ABAP programs that extracts the master data and creates the CSV files. The CSV files are then sent to the Ariba Procurement Solution. You can import master data using the following methods:

- Ariba Integration Toolkit
- Direct connectivity
- Mediated connectivity

Related Information

Integrating master data [page 85]

Transactional data

Transactional data includes information about purchase orders, invoices, receipts, and remittance advice. Transactional data is sent from the Ariba Procurement Solution to SAP ERP. However, transactional data for Remittances and the Purchase Order Header Status is always sent from SAP ERP to the Ariba Procurement Solution. You can import master data using the following methods:

- Ariba Integration Toolkit
- Direct connectivity
- Mediated connectivity
- Web services

You can export transactional data using either the File channel or Web services:

- File channel
  - The File channel uses the Data Transfer tool to export transactional data from the Ariba Procurement Solution to SAP ERP as CSV files for each integration event that you run.
  - The integration event extracts the data, converts the data to a CSV file and sends the file embedded in a MIME message to the Data Transfer tool. Based on the configurations, the Data Transfer tool downloads the CSV files to SAP ERP.

- Web services
  - If you are using the Web services to export transactional data, the SAP Process Integration middleware is used to export transactional data between the Ariba Procurement Solution and SAP ERP.
Related Information

Integrating transactional data using the file channel [page 171]
Integrating transactional data using the web services channel [page 200]
Integrating transactional data directly using the user interface [page 243]
Integrating transactional data using the mediated connectivity integration method [page 250]

Data integration methods

Buyers can import master data and export transactional data from SAP ERP to the Ariba Procurement Solution through the following methods:

- File-based integration
- Web services
- Direct connectivity
- Mediated connectivity

File-based integration

File-based integration enables you to import master data and transactional data using the Data Transfer Tool in the Ariba Integration Toolkit as follows:

- Ariba provides SAP transports that extracts master data from SAP ERP and writes it to a *.csv file. The *.csv files are then transferred to the Ariba Procurement Solution using the Ariba Integration Toolkit.
- You can export transactional data using the Data Transfer Tool. The Data Transfer tool downloads master data and transactional data from the Ariba Procurement Solution to the configured location on your file system based on the configuration specified in the BAT file. The data is stored as CSV files.

Related Information

File-based integration [page 13]
Web services based integration [page 15]
Direct connectivity integration [page 16]
Mediated connectivity integration [page 16]
The following graphic illustrates the integration of master data on file-based integration:

1. The ABAP program provided by Ariba downloads master data from SAP ERP in the form of CSV files.
2. The Data Transfer Tool then reads these CSV files and uploads data into the Ariba Procurement Solution.

Figure 1: Integration of Master data on file-based integration

### Related Information

Master data [page 11]

### Transactional data - file-based integration

The following graphic illustrates the integration of transactional data on file-based integration.
The Data Transfer Tool exports data from Ariba Procurement Solution in the form of CSV files.

2. The ABAP program provided by Ariba then read these CSV files and transfers data to SAP ERP.

3. The ABAP program provided by Ariba then retrieve the status of each export transaction from SAP ERP.

4. The Data Transfer tool reads these CSV files and uploads the status into Ariba Procurement Solution.

Web services based integration

Web services based integration allows integration of master data and transactional data. However, master data integration is not complete and allows you to import only some master data.

The following diagram illustrates the integration between Ariba Procurement Solution and SAP ERP using Web services.

The Web services provide for real-time integration of the Ariba Procurement Solution with SAP ERP using SAP Process Integration. The integration using SAP Process Integration is available by default. If you want to use any other application system through a SOA supporting middleware, then you are required to build your integration.

When an object is created in the Ariba Procurement Solution, which is configured to be sent to SAP ERP, the Web services generate a SOAP message based on the WSDL and dispatches it to a Web services server such as SAP Process Integration using the URL configured through the Ariba Administrator.
SAP Process Integration is configured to receive SOAP messages using the SOAP adapter. It transforms the data using the messaging mapping and converts the SOAP message to an SAP format (typically RFC). The RFC adapter then transmits the information to SAP ERP based on the configuration.

In SAP ERP, data is created using RFC/BAPI and the response (Success or failure) is sent back to SAP Process Integration, which again transforms the data and sends it back to the Ariba Procurement Solution through the Web services. The response is then updated in the respective Ariba Procurement Solution.

Related Information

Integrating transactional data using the web services channel [page 200]

Direct connectivity integration

Integrate master data and transactional data directly from SAP ERP to the Ariba Procurement Solution seamlessly without using the Ariba Integration Toolkit through the Direct Connectivity Integration method. The CSV files containing the master data is sent to the Ariba Procurement Solution through SOAP messages as follows:

- The master data is compressed in a ZIP file and embedded in a SOAP message header.
- A proxy is called with the request header.
- If there are failures while trying to connect to the Ariba Procurement Solution, the retry is based on the time duration you have configured.

Buyer administrators must configure the necessary parameters required to integrate master data. For more information, see Integrating master data using the direct connectivity integration method from SAP ERP [page 152].

Note

The Direct connectivity integration method for master data is available from Ariba Cloud Integration Release 4.0 and above. You can import transactional data using the Direct connectivity integration method from Ariba Cloud Integration Release 6.0 and above.

Related Information

Mediated connectivity integration [page 16]

Mediated connectivity integration

Buyers using the Ariba Procurement Solution integrated with SAP ERP can integrate master data and transactional data from SAP ERP to the Ariba Procurement Solution seamlessly using the SAP Process Integration
layer through the Mediated connectivity integration method. Buyers can securely download master data from SAP ERP to the Ariba Procurement Solution or Fieldglass system through the SAP Process Integration using SOAP messages. Buyers can use either the shared secret or client certificate-based authentication.

Buyer administrators must configure the necessary parameters required to integrate master data. For more information,

**Note**

The Mediated connectivity integration method for master data is available from Ariba Cloud Integration Release 5.0 and above. You can integrate transactional data using the Mediated connectivity integration method from Ariba Cloud Integration Release 6.0 and above.

**Related Information**

- Integrating master data using the mediated connectivity integration method [page 161]
- Integrating transactional data using the mediated connectivity integration method [page 250]
- Integrating transactional data directly using the user interface [page 243]
- Direct connectivity integration [page 16]

**Supported document transactions**

Ariba supports the following document transactions from SAP ERP and the Ariba Procurement Solution:

<table>
<thead>
<tr>
<th>Document sent from SAP ERP</th>
<th>Document sent from Ariba Procurement Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master Data for Supplier, User, Organization, Accounting, Cross/Application Configuration</td>
<td></td>
</tr>
<tr>
<td>Transactional Data</td>
<td>Purchase Orders, Change Purchase Orders, Cancel Purchase Orders</td>
</tr>
<tr>
<td>Purchase Order Status</td>
<td>Receipts</td>
</tr>
<tr>
<td>Receipt Status</td>
<td>Invoices</td>
</tr>
<tr>
<td>Invoice Status</td>
<td></td>
</tr>
<tr>
<td>Remittances</td>
<td>Expense Report</td>
</tr>
<tr>
<td>Contracts</td>
<td></td>
</tr>
</tbody>
</table>
Supported versions and integration landscapes

You can integrate Ariba Procurement Solution in the following SAP ERP versions:

- Ariba Cloud Integration release 8.0 and earlier: SAP ERP 6.0 (SAP_APPL 600 SPS02 and SAP_BASIS 700 SP18) (minimum requirement)
- Ariba Cloud Integration release 9.0 and higher: SAP ERP 6.0 (SAP_APPL 604 SPS05 and SAP_BASIS 701 SP05) to SAP EHP8 for SAP ERP 6.0 (SAP_APPL 618 SPS0004 and SAP_BASIS 750 SP0005)
- SAP S/4HANA, on-premise edition 1511
- SAP S/4HANA Finance
  - SAP Simple Finance, on-premise edition 1503
  - SAP S/4HANA Finance 1605

If you use the web services-based integration between Ariba Procurement Solution and SAP ERP, the following SAP Process Integration versions are supported:

- SAP Process Integration 7.1
- SAP Process Integration 7.3
- SAP Process Integration 7.31
- SAP Process Integration 7.4
- SAP Process Integration 7.5

**Note**

To prevent errors for the mandatory target fields containing a combination of NodeFunctions, CopyValue and Text Functions when using SAP Process Integration 7.5, refer to the following SAP Notes:

- SAP Note 2209925 - CopyValue function not getting correct output
- SAP Note 2185824 - Graphical Mapping combination - ‘CopyValue’, any Standard Functions from Text category and NodeFunctions give wrong output

Ariba recommends that you maintain separate installations of the integration for your Development and Production environments.

Installation

You must install the Ariba Integration Toolkit if you are using file-based integration. For more information, see the Ariba On Demand Solutions Integration Toolkit Guide.

Based on the solution you use, such as Ariba Procure-to-Pay and Ariba Procure-to-Order, or both, you must import the appropriate transports into SAP ERP. For more information on the installation and new features
available in the different Ariba Cloud Integration releases, see the *Installation Guide for Ariba Procurement Solution Integrated with SAP ERP*. 
Configuring SAP Process Integration

Configuring the System Landscape Directory [page 20]
Configuring transaction processes [page 24]
Configuring SAP Process Integration value mapping [page 27]
About configuring Integration Scenarios in the Integration Directory [page 28]
About Configuring the Integration Scenario [page 28]
Configuring SAP Process Integration for Web services integration [page 45]

Configuring the System Landscape Directory

Configure the System Landscape Directory (SLD) by importing the Ariba Procurement Solution product definition, and defining the systems involved in the integration. The SAP SLD is a central repository that holds the information on the systems and software in the customer environment. It consolidates the information about components actively available, and on which machines, instances, and clients they are available.

The Ariba Procurement Solution leverages all the advantages of the SLD in order to maintain the coherence of the system landscape as follows:

- It is defined as a product and component expressed in Common Information Interface (CIM) format importable in the SLD
- It uses a configuration using technical and business systems involved in the Ariba Procurement Solution integration and defined in the SLD.

You must make the following settings in the SLD:

- Import Ariba Procurement Product and Component definitions
- Define Technical and Business Landscape for the Ariba Procurement Solution
- Define Technical and Business Landscape for SAP ERP

The SAP Process Integration design packages (TPZ files) are available on https://connect.ariba.com in a ZIP file. You must download the ZIP file and extract the files for your required version.

How to import product and component definitions for the Ariba Procurement Solution

Context

Import the product and component definitions for the Ariba Procurement Solution that you require to work with SAP Process Integration, as a CIM file.
Procedure

1. From the SAP Process Integration main page, open the SLD administration home page.
2. Navigate to the Content section, and then click Import.
3. Choose the AribaOnDemand.SoftwareCatalog.zip file by clicking the Browse button, and then navigating to the file location.
4. Start the import.
5. In the Product Catalog, verify that the following product is available:
   ○ Vendor: ariba.com
   ○ Product: Ariba OnDemand
   ○ Version: Applicable version
6. Next, verify that the following components are available:
   ○ Vendor: ariba.com
   ○ Component: ARIBA_ONDEMAND_PURCHASEORDER, ARIBA_ONDEMAND_INVOICE,
     ARIBA_ONDEMAND_RECEIPT, ARIBA_ONDEMAND_REMITTANCE, ARIBA_ONDEMAND_ER_INVOICE,
     ARIBA_ONDEMAND_ADVANCE_PYMNT
   ○ Version: Applicable version

Results

The Ariba Procurement Solution is now recognized by the Integration Repository and Directory.

How to configure the Technical Landscape for the Ariba Procurement Solution

Procedure

1. Choose Technical Landscape from the main SLD page to open the Technical Landscape page.
3. In the Wizard, choose Third-Party as the Technical System Type to set up a third-party (non-SAP) application such as the Ariba Procurement Solution, and then click Next.
4. Specify the system details as follows:
   ○ System Name: Specify a user-defined system name here. For example, TS_BUYER.
   ○ Host Name: Enter ariba_buyer as the host name.
5. Click Next.
6. Associate the Ariba Procurement product and components to the system next. Choose the product Ariba Procurement, and then click Add. Ensure that you choose the components you want to implement.
7. Click Finish.
Next Steps

Set up the business landscape for the Ariba Procurement Solution using the Business Landscape Wizard.

How to configure the Business Landscape for the Ariba Procurement Solution

Procedure

1. Choose Business Landscape from the main SLD page to open the Business Landscape page.
2. Click New Business System to open the Business System Wizard.
3. In the wizard, enter the business system name, for example, BS_BUYER, and then click Next.
4. Choose Third-Party as the Technical System Type, and then click Next.
5. Associate the new business system with the required Technical System. From the dropdown, choose the technical system, TS_BUYER, created in the previous procedure, and then click Save.
   
   Note
   
   The Logical System Name is not mandatory.

6. Specify the installed products. Make sure that the product Ariba Procurement Solution and the related components to be implemented (for example, ARIBA_ONDEMAND_PURCHASEORDER) on the technical system are chosen, and then click Next.
7. Choose the role of the business system. The only available Business System Role for a third-party Business System is Application System. Choose your SAP Process Integration instance name from the dropdown to specify the Related Integration Server.
8. Click Finish.

How to configure the Technical Landscape for SAP ERP

Procedure

1. Choose Technical Landscape from the main SLD page to open the Technical Landscape page.
3. In the wizard, choose Web AS ABAP as the Technical System Type, and then click Next.
4. To specify the SAP ERP details enter the Web AS ABAP name (SID), installation number, and database host name, and then click Next.
5. Specify the Message Server and Central Application Server. Specify the Host Name, Message Port, and Logon Groups for the Message Server. Enter the Host Name and Instance Number for the Central Application Server, and then click Next.
6. Define additional application servers, if required. Specify the **Host Name** and **Instance Number** for the Application Server, and then click **Add**.

7. Click **Next**.

8. Define the Client. Specify the **Client Number**, and then click **Next**.

9. To add a **Logical System Name**, enter the system name, and then click **Add**. The logical system name is required for enabling the communication using IDocs between the SAP Process Integration and SAP ERP, and is used for routing documents in multi-ERP implementations.

10. Define software product and components. Choose the SAP ERP version installed on this technical system, as well as all other products installed on the system, and then click **Add**.

11. Click **Finish**.

**Next Steps**

Set up the business landscape for the SAP ERP system using the **Business Landscape Wizard**.

**How to configure the Business Landscape for SAP ERP**

**Procedure**

1. Choose **Business Landscape** from the main SLD page to open the **Business Landscape** page.

2. Click **New Business System** to open the **Business System Wizard**.

3. In the wizard, enter the business system name. For example, **BS_SAPR3_47**.

4. Click **Next**.

5. Choose **Web AS ABAP** as the **Technical System Type**, and then click **Next**.

6. Choose the technical system created in the previous procedure from the dropdown, to associate it with the business system, choose a client for the chosen technical system, and then click **Next**.

7. Specify the installed products. Choose the products that must be supported by this business system from the ones available from the associated technical system, to install them in the business system, and then click **Next**.

8. Choose the role of the business system. Choose **Application System** as the **Business System Role** and choose your SAP Process Integration instance name from the dropdown to specify the **Related Integration Server**. In the case of application systems, specify the integration server.

9. Click **Finish**.
Configuring transaction processes

After you have installed the SLD, do the following:

- Import the design package containing the transactions supported by the Ariba Procurement Solution.
- Configure SAP Process Integration connectivity to SAP ERP.
- Configure SAP Process Integration value mapping to provide the global configuration.
- Configure the different transactions in the Integration Directory by importing the scenarios from the Integration Repository.
- Configure the communication channels and the conditions for routing messages, for each supported transaction.

For more information on installing or upgrading the Ariba components for Ariba Procurement Solution, see the Installation Guide for Ariba Procurement Solution Integrated with SAP ERP.

**Note**

The names for objects in the Directory used in the configuration procedures are examples provided for illustration. You may use them or replace them with names and format corresponding to your system environment.

About importing the design package

The following design files are relevant for the transactions described:

<table>
<thead>
<tr>
<th>Transport File</th>
<th>Transactions</th>
</tr>
</thead>
</table>
| XI7_1_ARIBA_ONDEMAND_PURCHASEORDER_CIx_of_ariba.com.tpz | - Purchase Order
- Change Purchase Order
- Cancel Purchase Order
- Purchase Order Header Status Import |
| XI7_1_ARIBA_ONDEMAND_RECEIPT_CIx_of_ariba.com.tpz | Receipts (Receipt Export) |
| XI7_1_ARIBA_ONDEMAND_INVOICE_CIx_of_ariba.com.tpz | Invoices (Payment Export) |
| XI7_1_ARIBA_ONDEMAND_REMITTANCE_CIx_of_ariba.com.tpz | Remittance (Remittance Import) |
| XI7_1_ARIBA_ONDEMAND_ER_INVOICE_CIx_of_ariba.com.tpz | Expense Report (export) |
| XI7_1_ARIBA_ONDEMAND_BUDGET_MGMT_CIx_of_ariba.com.tpz | - Budget Check
- Requisition Submit
- Requisition Revert
- Requisition Derive
- Invoice Derive |
How to import the design package

Procedure

1. Copy the package file `XI7_1_ARIBA_ONDEMAND_<transaction>_CIX_of_ariba.com.tpz` to the import directory on the SAP Process Integration server, `*****\SYS\global\xi\repository_server\import`. For example, `D:\usr\sap\ARB\SYS\global\xi\repository_server\import`.
2. Open the Integration Builder design tool from the SAP Process Integration main page.
3. From the Tools menu, choose the Import Design Objects option.
4. Choose `XI7_1_ARIBA_ONDEMAND_<transaction>_CIX_of_ariba.com.tpz` from the available files list, and then click OK.

Results

After the import is complete, the component `ARIBA_ONDEMAND_<transaction>`, for CIx ariba.com appears in the components list of the Integration Builder design tool.

How to configure SAP ERP connectivity

Context

To enable communication between SAP Process Integration and SAP ERP using IDoc documents, you need to configure several parameters on the SAP Process Integration ABAP stack.

Procedure

1. Using Transaction SM59, create an RFC destination of type R/3 to connect to SAP ERP.
2. Using Transaction IDX1, create a Port for the new RFC destination. For example, for the SAP ERP system D47 client 800, you must use specify following parameters to create the port:

- **Port**: SAPD47.
- **Client**: 800.
- **RFC Destination**: Specify the destination created in SM59 here.

**Note**
Enter the password in UPPER case.

---

### How to assign Business Systems in the Integration Directory

**Context**

The Integration Directory configuration uses the information on the Technical and Business systems from the SLD. You must assign the Business Systems defined and involved in the integration setup.

**Procedure**

1. Open the **Configuration: Integration Builder** from the **SAP Process Integration** main page.
2. In the **Objects** tab of the **Integration Builder**, navigate to **Service Without Party** ➔ **Business System** ➔
3. From the context menu for **Business System**, choose **Assign Business System** to start the **Assign Business System wizard**.
4. Click **Continue** in the first and second windows. Leave the **Party** field in the second window blank.
5. In the third window, **Select Business Systems**, choose the business systems **BS_BUYER** and **BS_SAPR3_46C**, created earlier, from the list. For more information about creating business systems, see [Configuring the System Landscape Directory](page 20).

   In the case of Multi-ERP implementation, you must choose all the Business Systems representing your SAP ERP instances involved in the integration.

   Also, uncheck **Create Communication Channel Automatically** to avoid the creation of unwanted channels.
6. Click **Finish**.

---

### How to create Communication Channels for SAP ERP

**Context**

Create the communication channels used for SAP ERP connectivity. You must create the RFC Receiver communication channels from the Integration Builder, to enable communication using the RFC invocation on the
remote SAP ERP system. This channel will be used by almost all the scenarios and are common to all of them. Therefore, you only need to create them once, for each Business System corresponding to your SAP ERP.

**Procedure**

1. In the **Objects** tab of the **Integration Builder**, navigate to **Service Without Party** ➔ **Business System** ➔ **BS_SAPR3_46C** ➔ **Communication Channel**.
2. From the context menu for **Communication Channel**, choose **New**.
3. In the **Create Object** window, enter **CC_RFC_Receiver** as the name of the communication channel, in the **Communication Channel** field, and then click **Create**.
4. From the **Parameters** tab of the **Edit Communication Channel** window, click the icon next to the **Adapter Type** field to display the list of adapter types. From the adapter type list window, choose the following adapter and click **OK**:
   - Name: RFC
   - Namespace: http://sap.com/xi/XI/System
   - Software Component Version: SAP BASIS 7.10 of SAP Process Integration 7.1 and SAP Basis 7.11 for SAP Process Integration 7.1 EHP1
5. After choosing the adapter, click the **Receiver** option, to choose the corresponding type.
6. In the **Adapter Engine** field, choose the **Integration Server** or **Non-Central Adapter Engine**, depending on your configuration.
7. Edit the other parameters to match your SAP ERP system (**Application Server**, **System Number**, **Authentication Mode**, **Logon User**, **Logon Password**, **Logon Language**, **Logon Client**).  
   **Note**
   
   Enter the password in UPPER case.
8. Edit the **Maximum Connections** option according to the expected load.
9. Save and activate the communication channel.

   In the case of Multi-ERP implementation, you must create one **CC_RFC_Receiver** channel per Business System, representing each of the SAP ERP instances involved in the integration.

**Configuring SAP Process Integration value mapping**

Configure SAP Process Integration value mapping for supporting facilities such as IDoc Payload and standard code conversions.
Configuring global variables

There are many parameters that are global for each of the supported scenarios when SAP Process Integration acts as the integration platform between SAP ERP and the Ariba Procurement Solution. This section describes the steps to configure global variables.

In the value mappings, use the following parameters for Global Variables:

- **Source Values:**
  - **Agency:** XI_Global
  - **Scheme:** GlobalVariable

- **Target Values:**
  - **Agency:** XI
  - **Scheme:** GlobalVariable

### Key value pairs

<table>
<thead>
<tr>
<th>XI_Global</th>
<th>XI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AribaBuyerTimeZone</td>
<td>+5:30</td>
<td>Time zone of the Ariba Procurement Solution.</td>
</tr>
</tbody>
</table>

The XI column in the above table indicates a sample value.

### About configuring Integration Scenarios in the Integration Directory

The following sections describe how to import and configure the integration scenarios in the SAP Process Integration Directory for the supported transactions. Use an integration scenario from the Integration Repository as a template for configuring the corresponding transaction in the Integration Directory. The Integration Scenarios are part of the Design package, which has to be imported into the Integration Repository first.

**Note**

Do not activate the Change Lists feature of Integration Builder Configuration Tool until all the configuration steps are completed.

### About Configuring the Integration Scenario

You need to configure the integration scenario for the all transactions in the Integration Directory to be able to run each transaction. All the steps of a configuration scenario section must be completed to fully configure a scenario.
If you are using a SAP NetWeaver PI version that is higher than SAP BASIS 7.10, then you must ensure that you edit the software component version for the communication channel template to avoid receiving the following error: "Software component version with name SAP BASIS 7.10 not found. See error log for details." For more information, see How to configure the software component version for communication channel templates [page 37].

Ensure that the following design packages corresponding to the different transactions and functions are imported into your system before you configure an Integration Scenario.

Table 1: Integration Scenario Design Packages

<table>
<thead>
<tr>
<th>Integration Scenario</th>
<th>Design Package</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requisition</td>
<td>XI7_1_ARIBA_ONDEMAND_BUDGET_MGMT_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Requisition Revert</td>
<td>XI7_1_ARIBA_ONDEMAND_BUDGET_MGMT_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Purchase Order</td>
<td>XI7_1_ARIBA_ONDEMAND_PURCHASEORDER_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Change Purchase Order</td>
<td>XI7_1_ARIBA_ONDEMAND_PURCHASEORDER_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Close Purchase Order</td>
<td>XI7_1_ARIBA_ONDEMAND_PURCHASEORDER_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Cancel Order</td>
<td>XI7_1_ARIBA_ONDEMAND_PURCHASEORDER_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Purchase Order Header Status</td>
<td>XI7_1_ARIBA_ONDEMAND_PURCHASEORDER_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Receipt</td>
<td>XI7_1_ARIBA_ONDEMAND_RECEIPT_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Invoice</td>
<td>XI7_1_ARIBA_ONDEMAND_INVOICE_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Remittance</td>
<td>XI7_1_ARIBA_ONDEMAND_REMITTANCE_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Expense Report</td>
<td>XI7_1_ARIBA_ONDEMAND_ER_INVOICE_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Budget Check</td>
<td>XI7_1_ARIBA_ONDEMAND_BUDGET_MGMT_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Funds Derivation</td>
<td>XI7_1_ARIBA_ONDEMAND_BUDGET_MGMT_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Advance Payment Request</td>
<td>XI7_1_ARIBA_ONDEMAND_ADVANCE_PYMNT_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Create Service Entry Sheet</td>
<td>XI7_1_ARIBA_ONDEMAND_SRVENTRYSHEET_CIx_of_ariba.com.tpz</td>
</tr>
<tr>
<td>Service Entry Sheet Response</td>
<td>XI7_1_ARIBA_ONDEMAND_SRVENTRYSHEET_CIx_of_ariba.com.tpz</td>
</tr>
</tbody>
</table>

The following steps assume that the different Business Systems are defined in the SLD. For more information about configuring the SLD, see Configuring the System Landscape Directory [page 20].

Follow these procedures to configure an Integration Scenario:

1. Transfer the Integration Scenario from the Integration Repository to the Integration Directory. See How to transfer the Integration Scenario [page 30]
2. Configure the Integration Scenario. See Configuring the Integration Scenario [page 31].
3. Activate the new Integration Scenario. See Activating the Integration Scenario [page 40].
How to transfer the Integration Scenario

Context

The first step to configure the integration scenario for a transaction is to transfer to the Integration Directory. The integration scenario is loaded into the Integration Repository when the Design package is imported.

Procedure

1. From the SAP Process Integration main page, choose Integration Directory Integration Builder, and then log in to the Configuration: Integration Builder application.
2. In the Integration Builder menu, choose Tools Apply Model from ES Repository.
3. In the first screen of the Transfer Model from ES Repository wizard, choose Process Integration Scenario.
4. Click the Name field, click the Display Input Help dropdown and then choose the Input Help.
5. Choose the following options for the Process Integration Scenario from the Enterprise Services Repository screen:
   - Name
   - Namespace
   - Software Component Version

     For information about the values that you must specify for these fields, see Integration Scenario Parameter Values [page 41].

6. Click Continue.
7. If required, change the name of the scenario for the Integration Directory.
8. Click Finish.

    After the transfer is completed, the Integration Scenario Configurator window appears.

9. (Required only for transferring the budget integration scenario) Repeat step 3 through step 6. Choose the following options for the Process Integration Scenario from the Enterprise Services Repository screen:

   - Name: AR_PurchaseRequisition_Send
   - Namespace: urn:ariba.com:xi:OnDemand:BudgetCheck
   - Software Component Version: ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com
Next Steps

Configure the integration scenario. See Configuring the Integration Scenario [page 31].

Related Information

About Configuring the Integration Scenario [page 28]

Configuring the Integration Scenario

To configure the integration scenario, you must perform the following tasks:

- How to choose the model for the Integration Scenario [page 31]
- How to assign components for the Integration Scenario [page 32]
- How to configure the sender and receiver relations for individual connections for the integration scenario [page 33]
- How to generate objects for the Integration Scenario [page 35]
- How to configure interface determination settings for the Integration Scenario [page 36]
- How to configure the software component version for communication channel templates [page 37]
- How to enable support for attachments in the Integration Scenario [page 39]

How to choose the model for the Integration Scenario

Procedure

In the Model Configurator window, click Select Component View, and then click Apply to choose the unique component view that is chosen by default.

Next Steps

After choosing the component view, you must assign components to the application components of the chosen integration scenario. See How to assign components for the Integration Scenario [page 32].
How to assign components for the Integration Scenario

Prerequisites

- Transfer the Integration scenario. See How to transfer the Integration Scenario [page 30].
- Choose the model for the scenario. See How to choose the model for the Integration Scenario [page 31].

Context

Perform the following procedure to assign the business systems to the Ariba Procurement Solution and SAP ERP components:

Procedure

1. In the Model Configurator, click Assign Component.
2. In the Assign Components window, choose the Business System Components for A2A tab to assign components for the Ariba Network.
3. Click Insert Line (+) to add a component entry in the table if no component entry line is available.
4. Click the row under the Communication Component column, click the Value List dropdown, and choose Value List or <F4>. Choose the communication component name of the business system for your SAP ERP, from the selection list and click Apply.

   **Note**
   
   If you have multiple ERP systems, add communication components defined for each of your SAP ERP by repeating step 3 through step 4.

5. Click Next Role (the right arrow) to go to the next screen on the window.
6. Click Insert Line (+) to add a component entry in the table if no component entry line is available.
7. Click the row under the Communication Component column, click the Value List dropdown, and choose Value List <F4>. Choose the communication component name of the business system for the Ariba Procurement Solution from the selection list and click Apply.
8. Click Save Settings.
Next Steps

Configure the sender and receiver relations for individual connections for the integration scenario. See How to configure the sender and receiver relations for individual connections for the integration scenario [page 33].

Related Information

About Configuring the Integration Scenario [page 28]

How to configure the sender and receiver relations for individual connections for the integration scenario

Prerequisites

- Assign the components for the integration scenario. See How to assign components for the Integration Scenario [page 32].
- Create the communication channels for the SAP ERP. See How to create Communication Channels for SAP ERP [page 26].

Context

To activate the assignment of sender and receiver communication components at connection level for the integration scenario, perform the following steps:

Procedure

1. In the Model Configurator, click Configure Connections.
2. In the Connections from Component Assignment tab in the Configure Connection window, click the empty Communication Channel cell corresponding to the business system for the Ariba Network in the Sender Business System Components table.
3. On the top left corner of the screen, click New Communication Channel and then choose Create Communication Channel with Template to start the Create Communication Channel wizard.
4. Click Continue.
5. On the next page, click the Name field, click the Display Input Help dropdown, and choose Input Help. Choose the following Communication Channel Template:
   - Name
6. Click **Apply** and then click **Continue** and if required, modify the channel name.
7. Click **Finish** and then click **Close**.
8. In the **Connections from the Component Assignment** tab in the **Configure Connection** window, click on the empty **Communication Channel** cell corresponding to the business system for SAP ERP in the **Receiver Business System Components** table.
9. Click **Value List** (F4) from the dropdown, and then choose **RFC communication channel** from the **Communication Channel for Receiver Communication Component**.
10. Click **Apply** and then click **Next Connection**.

For configurations other than requisitions and funds derivation, go to step 12.

11. To configure the sender and receiver relations for advance payments, requisitions, and funds derivation, you must repeat step 2 through step 10 with the following changes:
   - For advance payments:
     - In step 5, replace the communication channel template name
       `CCT_AdvancePaymentRequest_Create` with `CCT_AdvancePaymentRequest_Cancel`.
     - In step 6, replace the channel name `CCT_AdvancePaymentRequest_Create` with `CCT_AdvancePaymentRequest_Cancel`.
   - For requisitions:
     - In step 5, replace the communication channel template name `CCT_ReqSubmit_Sender` with `CCT_ReqWithdraw_Sender`.
     - In step 6, replace the channel name `CCT_ReqSubmit_Sender` with `CCT_ReqWithdraw_Sender`.
   - For funds derivation:
     - In step 5, replace the communication channel template name `CCT_ReqFMDerive_Sender` with `CCT_InvFMDerive_Sender`.
     - In step 6, replace the channel name `CCT_ReqFMDerive_Sender` with `CCT_InvFMDerive_Sender`.
12. Click **Activate All Connections** and then click **OK**.

**Next Steps**

Generate objects for the integration scenario. See *How to generate objects for the Integration Scenario [page 35]*.

**Related Information**

*About Configuring the Integration Scenario [page 28]*
How to generate objects for the Integration Scenario

Prerequisites

Configure sender and receiver relations for the integration scenario. See How to configure the sender and receiver relations for individual connections for the integration scenario [page 33].

Context

Generating the objects creates the configuration objects if they do not yet exist. If they already exist, they are reused and enhanced, if required. You can also simulate generation of objects for test purposes.

Procedure

1. In the Model Configurator, click Create Configuration Object.
2. In the Create Configuration Object window, do the following:
   ○ Click Generation in the General section.
   ○ Check Receiver Determination and Interface Determination in the Scope of Generation section.
3. Click Create New in the Change List for Generated Objects section. The text box already contains an object name that you can use or modify.
4. Click Start to begin generating objects.
5. After the object generation is complete, click Apply to save the new configuration settings. Close the log that is created and save it, if required.

Next Steps

Configure interface determination settings for the integration scenario. See How to configure interface determination settings for the Integration Scenario [page 36].

Related Information

About Configuring the Integration Scenario [page 28]
How to configure interface determination settings for the Integration Scenario

Prerequisites

Assign the communication channels for the integration scenario. See How to create Communication Channels for SAP ERP [page 26].

Context

To configure the interface to process the data message for the Integration scenario, perform the following steps:

Procedure

1. Expand Interface Determination, and then click | Name of the Business System for Ariba Network | MIOut_Sync_WSDL_<transaction> | Name of the Business System for SAP ERP |
2. In the Receiver Interfaces section of the Edit Interface Determination window, click the operation mapping value under the Operation Mapping column. Click the Value List dropdown and choose Value List.
3. Choose the appropriate operation mapping and click Apply.
4. The Multiplicity column displays the value 0..unbounded.
5. Save and activate the interface determination settings.
   If the integration scenario includes multiple Interface Determinations then you must repeat the above procedure for each interface.

   i Note
   Configuring the Interface Determination is a mandatory step if you configure the mappings.

Next Steps

Configure the software component version for communication channel templates. See How to configure the software component version for communication channel templates [page 37].

Related Information

About Configuring the Integration Scenario [page 28]
How to configure the software component version for communication channel templates

Context

If you are using a SAP Process Integration version that is higher than SAP BASIS 7.10, then you must ensure that you edit the communication channel template to avoid receiving an error.

Important: Ensure that you have the necessary options enabled for the Object Properties on the Software Component Version page that allows you to edit the communication channel template.

Procedure

1. Go to the Enterprise Service Builder and edit each of the following Software Component Versions to allow you to modify the objects:
   - ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com
   - ARIBA_ONDEMAND_ER_INVOICE CIx of ariba.com
   - ARIBA_ONDEMAND_INVOICE CIx of ariba.com
   - ARIBA_ONDEMAND_PURCHASEORDER CIx of ariba.com
   - ARIBA_ONDEMAND_RECEIPT CIx of ariba.com
   - ARIBA_ONDEMAND_REMITTANCE CIx of ariba.com
   - ARIBA_ONDEMAND_ADVANCE_PYMT CIx of ariba.com
   - ARIBA_ONDEMAND_SRVENTRYSHEET CIx of ariba.com

2. Double click the required communication channel template. The Display Communication Channel Template page appears.

3. In the Parameters tab, click the Display Input Help button in the Adapter Type field. The Choose Adapter Metadata page appears.

4. Choose the common Basis version for each of the following communication channel templates:

<table>
<thead>
<tr>
<th>Communication Channel Template</th>
<th>Adapter Type</th>
<th>Namespace</th>
<th>Software Component Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com</td>
<td>SOAP</td>
<td><a href="http://sap.com/">http://sap.com/</a> XI/ XI/ System</td>
<td>Choose the common Basis version. For example, SAP BASIS 7.40.</td>
</tr>
<tr>
<td>CCT_BudgetCheck_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARIBA_ONDEMAND_ER_INVOICE CIx of ariba.com</td>
<td>SOAP</td>
<td><a href="http://sap.com/">http://sap.com/</a> XI/ XI/ System</td>
<td>Choose the common Basis version. For</td>
</tr>
<tr>
<td>CC_Ondemand_ER_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Communication Channel Template</td>
<td>Adapter Type</td>
<td>Namespace</td>
<td>Software Component Version</td>
</tr>
<tr>
<td>------------------------------------------------------------------------------------------------</td>
<td>--------------</td>
<td>-----------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ARIBA_ONDEMAND_INVOICE CIx of ariba.com</td>
<td>SOAP</td>
<td><a href="http://sap.com/xi/XI/System">http://sap.com/xi/XI/System</a></td>
<td>Choose the common Basis version. For example, SAP BASIS 7.40.</td>
</tr>
<tr>
<td>CCT_Invoice_WSDL_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARIBA_ONDEMAND_PURCHASEORDER CIx of ariba.com</td>
<td>SOAP</td>
<td><a href="http://sap.com/xi/XI/System">http://sap.com/xi/XI/System</a></td>
<td>Choose the common Basis version. For example, SAP BASIS 7.40.</td>
</tr>
<tr>
<td>CCT_CancelPurchaseOrder_WSDL_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCT_ChangePurchaseOrder_WSDL_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCT_PurchaseOrder_WSDL_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCT_PurchaseOrderHeader_WSDL_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCT_ClosePurchaseOrder_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARIBA_ONDEMAND_RECEIPT CIx of ariba.com</td>
<td>SOAP</td>
<td><a href="http://sap.com/xi/XI/System">http://sap.com/xi/XI/System</a></td>
<td>Choose the common Basis version. For example, SAP BASIS 7.40.</td>
</tr>
<tr>
<td>CCT_Receipts_WSDL_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARIBA_ONDEMAND_REMITTANCE CIx of ariba.com</td>
<td>SOAP</td>
<td><a href="http://sap.com/xi/XI/System">http://sap.com/xi/XI/System</a></td>
<td>Choose the common Basis version. For example, SAP BASIS 7.40.</td>
</tr>
<tr>
<td>CCT_RemittanceImport_WSDL_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com</td>
<td>SOAP</td>
<td><a href="http://sap.com/xi/XI/System">http://sap.com/xi/XI/System</a></td>
<td>Choose the common Basis version. For example, SAP BASIS 7.40.</td>
</tr>
<tr>
<td>CCT_BudgetCheck_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com</td>
<td>SOAP</td>
<td><a href="http://sap.com/xi/XI/System">http://sap.com/xi/XI/System</a></td>
<td>Choose the common Basis version. For example, SAP BASIS 7.40.</td>
</tr>
<tr>
<td>CCT_ReqFMDerive_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Communication Channel Template</td>
<td>Adapter Type</td>
<td>Namespace</td>
<td>Software Component Version</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>--------------</td>
<td>------------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>ARIBA_ONDEMAND_ADVANCE_PYMNT CIx of ariba.com</td>
<td>SOAP</td>
<td><a href="http://sap.com/xi/XI/System">http://sap.com/xi/XI/System</a></td>
<td>Choose the common Basis version. For example, SAP BASIS 7.40.</td>
</tr>
<tr>
<td>CCT_AdvancePaymentRequest_Create_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCT_AdvancePaymentRequest_Cancel_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCT_AdvPymntRemittanceImport_WSDL_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARIBA_ONDEMAND_SRVENTRIESHEET CIx of ariba.com</td>
<td>SOAP</td>
<td><a href="http://sap.com/xi/XI/System">http://sap.com/xi/XI/System</a></td>
<td>Choose the common Basis version. For example, SAP BASIS 7.40.</td>
</tr>
<tr>
<td>CCT_ServiceEntrySheet_WSDL_Sender</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5. Save and activate the changes.

**Next Steps**

Activate the integration scenario. See Activating the Integration Scenario [page 40].

**Related Information**

Configuring the Integration Scenario [page 31]

**How to enable support for attachments in the Integration Scenario**

**Context**

To enable support for attachments with transactional documents such as, purchase orders and invoices, perform the following steps:

**Note**

If you are using a SAP Process Integration version that is higher than SAP BASIS 7.10, then you must ensure that you edit the communication channel template to avoid receiving an error.
Ensure that you have the necessary options enabled for the **Object Properties** on the page **Software Component Version** that allows you to edit the communication channel template.

**Procedure**

1. Navigate to the **Integration Builder** in your SAP Process Integration system.
2. Navigate to **Business System BS_ONDEMAND** and double-click the communication channel.
3. Check **Keep Attachments** under the tab **Parameters General**.
4. Save and activate the communication channel.

**Related Information**

- About attachments with purchase orders and invoices in the Ariba Procurement Solution integrated with SAP ERP [page 237]
- Configuring the Integration Scenario [page 31]

### Activating the Integration Scenario

Using the **Change List** tab on the main application screen, activate all the Change Lists containing the objects created and modified during the scenario setup.

This completes the configuration of the integration scenario in SAP Process Integration.

**Related Information**

- About Configuring the Integration Scenario [page 28]
## Integration Scenario parameter values

The following table contains information about the values you need to specify when you are configuring an integration scenario for the different transactions:

### Table 2: Integration Scenario parameter values

<table>
<thead>
<tr>
<th>Transaction</th>
<th>Integration Scenario</th>
<th>Communication Channel Template</th>
<th>Values for configuring communication channels</th>
</tr>
</thead>
</table>
| Submit Requisition           | ● AR_PurchaseRequisition_Send  
   ● urn:ariba.com:x:i:OnDemand:PurchaResRequisition  
   ● ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com | ● CCT_ReqSubmit_Sender  
   ● urn:ariba.com:x:i:OnDemand:PurchaResRequisition  
   ● ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com | ● urn:ariba.com:x:i:OnDemand:PurchaseRequisition  
   ● MIOut.Sync_WSDL_ReqSubmit |
| Cancel/Revert Requisition    | ● AR_PurchaseRequisition_Send  
   ● urn:ariba.com:x:i:OnDemand:PurchaResRequisition  
   ● ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com | ● CCT_ReqWithdraw_Sender  
   ● urn:ariba.com:x:i:OnDemand:PurchaResRequisition  
   ● ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com | ● urn:ariba.com:x:i:OnDemand:PurchaseRequisition  
   ● MIOut.Sync_WSDL_ReqWithdraw |
| Purchase Order               | ● AR_PurchaseOrder_Send  
   ● urn:ariba.com:x:i:OnDemand:CreatePurchaseOrder  
   ● ARIBA_ONDEMAND_PURCHASEORDER, CIx of ariba.com | ● CCT_PurchaseOrder_WSDL_Sender  
   ● urn:ariba.com:x:i:OnDemand:CreatePurchaseOrder  
   ● ARIBA_ONDEMAND_PURCHASEORDER, CIx of ariba.com | ● urn:ariba.com:x:i:OnDemand:CreatePurchaseOrder  
   ● MIOut.Sync_WSDL_WSPurchaseOrderPushRequestMessage |
| Change Purchase Order        | ● AR_ChangePurchaseOrder_Send  
   ● urn:ariba.com:x:i:OnDemand:ChangePurchaseOrder  
   ● ARIBA_ONDEMAND_PURCHASEORDER, CIx of ariba.com | ● CCT_ChangePurchaseOrder_WSDL_Sender  
   ● urn:ariba.com:x:i:OnDemand:ChangePurchaseOrder  
   ● ARIBA_ONDEMAND_PURCHASEORDER, CIx of ariba.com | ● urn:ariba.com:x:i:OnDemand:ChangePurchaseOrder  
   ● MIOut.Sync_WSDL_WSPurchaseOrderChangePushRequestMessage |
<table>
<thead>
<tr>
<th>Transaction</th>
<th>Integration Scenario transfer</th>
<th>Communication Channel Template</th>
<th>Values for configuring communication channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Close Purchase Order</td>
<td>● AR_ChangePurchaseOrder_Send</td>
<td>● CCT_ClosePurchaseOrder_Sender</td>
<td>● urn:ariba.com:xii:OnDemand:ChangePurchaseOrder ariba.com</td>
</tr>
<tr>
<td></td>
<td>● urn:ariba.com:xii:OnDemand:ChangePurchaseOrder ariba.com</td>
<td>● ARIBA_ONDEMAND_PURCHASEORDER, CIx of ariba.com</td>
<td>● ARIBA_ONDEMAND_PURCHASEORDER, CIx of ariba.com</td>
</tr>
<tr>
<td>Cancel Purchase Order</td>
<td>● AR_CancelPurchaseOrder_Send</td>
<td>● CCT_CancelPurchaseOrder_WSDL_Sender</td>
<td>● urn:ariba.com:xii:OnDemand:CancelPurchaseOrder ariba.com</td>
</tr>
<tr>
<td></td>
<td>● urn:ariba.com:xii:OnDemand:CancelPurchaseOrder ariba.com</td>
<td>● ARIBA_ONDEMAND_PURCHASEORDER, CIx of ariba.com</td>
<td>● ARIBA_ONDEMAND_PURCHASEORDER, CIx of ariba.com</td>
</tr>
<tr>
<td>Purchase Order Header Status</td>
<td>● AR_PurchaseOrderHeader_Send</td>
<td>● CCT_PurchaseOrderHeader_WSDL_Sender</td>
<td>● urn:ariba.com:xii:OnDemand:PurchaseOrderHeaderStatus ariba.com</td>
</tr>
<tr>
<td></td>
<td>● urn:ariba.com:xii:OnDemand:PurchaseOrderHeaderStatus ariba.com</td>
<td>● ARIBA_ONDEMAND_PURCHASEORDER, CIx of ariba.com</td>
<td>● ARIBA_ONDEMAND_PURCHASEORDER, CIx of ariba.com</td>
</tr>
<tr>
<td>Receipt</td>
<td>● AR_Receipt_Send</td>
<td>● CCT_Receipt_WSDL_Sender</td>
<td>● urn:ariba.com:xii:OnDemand:Receipt ariba.com</td>
</tr>
<tr>
<td></td>
<td>● urn:ariba.com:xii:OnDemand:Receipt ariba.com</td>
<td>● ARIBA_ONDEMAND_CREATE_RECEIPT, CIx of ariba.com</td>
<td>● ARIBA_ONDEMAND_CREATE_RECEIPT, CIx of ariba.com</td>
</tr>
<tr>
<td></td>
<td>● ARIBA_ONDEMAND_CREATE_RECEIPT, CIx of ariba.com</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transaction</td>
<td>Integration Scenario transfer</td>
<td>Communication Channel Template</td>
<td>Values for configuring communication channels</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------</td>
<td>--------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>Invoice</td>
<td>● AR_Invoice_Send</td>
<td>● CCT_Invoice_WSDL_Sender</td>
<td>• urn:ariba.com:xi:OnDemand:Invoice</td>
</tr>
<tr>
<td></td>
<td>● urn:ariba.com:xii:OnDemand:Invoice</td>
<td>• urn:ariba.com:xii:OnDemand:Invoice</td>
<td>• MI_Out_Sync_WSDL_WSInvoicePushRequestMessage</td>
</tr>
<tr>
<td></td>
<td>● ARIBA_ONDEMAND_INVOICE, CIx of ariba.com</td>
<td>• urn:ariba.com:xii:OnDemand:Invoice</td>
<td>• MI_Out_Sync_WSDL_WSInvoicePushRequestMessage</td>
</tr>
<tr>
<td>Remittance</td>
<td>● AR_RemittanceImport_Send</td>
<td>● CCT_RemittanceImport_WSDL_Sender</td>
<td>• urn:ariba.com:xi:OnDemand:Remittance</td>
</tr>
<tr>
<td></td>
<td>● urn:ariba.com:xii:OnDemand:Remittance</td>
<td>• urn:ariba.com:xii:OnDemand:Remittance</td>
<td>• MI_Out_Sync_WSDL_RemittanceImportRequestMessage</td>
</tr>
<tr>
<td></td>
<td>● ARIBA_ONDEMAND_ICREATE_REMITTANCE, CIx of ariba.com</td>
<td>• urn:ariba.com:xii:OnDemand:Remittance</td>
<td>• MI_Out_Sync_WSDL_RemittanceImportRequestMessage</td>
</tr>
<tr>
<td>Expense Report</td>
<td>● AR_OnDemand_ER_Send</td>
<td>● CCT_OnDemand_ER_Sender</td>
<td>• urn:ariba.com:xi:OnDemand:ExpenseReport</td>
</tr>
<tr>
<td></td>
<td>● ARIBA_ONDEMAND_ER_INVOICE, CIx of ariba.com</td>
<td>• urn:ariba.com:xii:OnDemand:ExpenseReport</td>
<td>• MI_Out_Sync_WSDL_ExpenseReportPushRequestMessage</td>
</tr>
<tr>
<td>Create Advance Payment Request</td>
<td>● AR_AdvancePaymentRequest_Send</td>
<td>● CCT_AdvancePaymentRequest_Create_Sender</td>
<td>• urn:ariba.com:xi:OnDemand:AdvancePayment</td>
</tr>
<tr>
<td></td>
<td>● urn:ariba.com:xii:OnDemand:AdvancePayment</td>
<td>● CCT_AdvancePaymentRequest_Cancel_Sender</td>
<td>• MI_Out_Sync_WSDL_AdvancePaymentRequest_Create</td>
</tr>
<tr>
<td></td>
<td>● ARIBA_ONDEMAND_ADVANCE_PYMNT_CIx of ariba.com</td>
<td>● urn:ariba.com:xii:OnDemand:AdvancePayment</td>
<td>• MI_Out_Sync_WSDL_AdvancePaymentRequest_Create</td>
</tr>
<tr>
<td>Transaction</td>
<td>Integration Scenario transfer</td>
<td>Communication Channel Template</td>
<td>Values for configuring communication channels</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Advance Payment Remittance Request</td>
<td>• AR_AdvancePaymentRemittance_Send</td>
<td>• CCT_AdvPyntRemittanceImport_WS DL_Sender</td>
<td>• urn:ariba.com:xi:OnDemand:AdvancePaymentRemittanceRequestMessage</td>
</tr>
<tr>
<td></td>
<td>• urn:ariba.com:xi:OnDemand:AdvancePaymentRemittance</td>
<td>• urn:ariba.com:xi:OnDemand:AdvancePaymentRemittance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• ARIBA_ONDEMAND_ADVANCE_PYMT_CIx of ariba.com</td>
<td>• ARIBA_ONDEMAND_ADVANCE_PYMT_CIx of ariba.com</td>
<td></td>
</tr>
<tr>
<td>Service Entry Sheet</td>
<td>• AR_ServiceEntrySheet_Send</td>
<td>• CCT_ServiceEntrySheet_WSDL_Sender</td>
<td>• urn:ariba.com:xi:OnDemand:CreateServiceEntrySheet</td>
</tr>
<tr>
<td></td>
<td>• urn:ariba.com:xi:OnDemand:CreateServiceEntrySheet</td>
<td>• urn:ariba.com:xi:OnDemand:CreateServiceEntrySheet</td>
<td>• MIOut.Sync_WSDL_ServiceEntrySheetPushRequestMessage</td>
</tr>
<tr>
<td></td>
<td>• ARIBA_ONDEMAND_SRVENTRYSHEET_CIx of ariba.com</td>
<td>• ARIBA_ONDEMAND_SRVENTRYSHEET_CIx of ariba.com</td>
<td></td>
</tr>
<tr>
<td>Service Entry Sheet Response</td>
<td>• AR_SES_Response_ReceiveProxy</td>
<td>• CCT_ServiceEntrySheetResponse_WSDL_Receiver</td>
<td>• urn:ariba.com:xi:OnDemand:ServiceEntrySheetResponse</td>
</tr>
<tr>
<td></td>
<td>• ARIBA_ONDEMAND_SRVENTRYSHEET_CIx of ariba.com</td>
<td>• ARIBA_ONDEMAND_SRVENTRYSHEET_CIx of ariba.com</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Integration Scenario Parameter Values

<table>
<thead>
<tr>
<th>Function</th>
<th>Integration Scenario Transfer</th>
<th>Communication Channel Template</th>
<th>Values for configuring communication channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Budget Check</td>
<td>• AR_BudgetCheck_Send</td>
<td>• CCT_BudgetCheck_Sender</td>
<td>• urn:ariba.com:xi:OnDemand:BudgetCheck</td>
</tr>
<tr>
<td></td>
<td>• urn:ariba.com:xi:OnDemand:BudgetCheck</td>
<td>• urn:ariba.com:xi:OnDemand:BudgetCheck</td>
<td>• MIOut.Sync_WSDL_BudgetCheck</td>
</tr>
<tr>
<td></td>
<td>• ARIBA_ONDEMAND_BUDGET_MGMT_CIx of ariba.com</td>
<td>• ARIBA_ONDEMAND_BUDGET_MGMT_CIx of ariba.com</td>
<td></td>
</tr>
</tbody>
</table>
### Configuring SAP Process Integration for Web services integration

This section provides information on the activities that you need to perform to enable Ariba Procurement Solution to integrate with SAP Process Integration using the Web services channel.

### How to generate keys for signing SAP Process Integration messages

**Context**

In SAP Process Integration, you can manage the certificates from the key storage of the Visual administrator.

**Procedure**

1. To start the visual administrator, go to the link: `/usr/sap/DVEBMGS00/j2ee/admin` and run the script `go.bat` and click service `KeyStorage`.
2. In the `Runtime` tab, under the `Entry` section for `Generate a pair of keys (public/private)`, click `Create`.
3. In the `Key and Certificate Generation` screen, enter values for all the fields as required. You can also modify existing values.

---

<table>
<thead>
<tr>
<th>Function</th>
<th>Integration Scenario Transfer</th>
<th>Communication Channel Template</th>
<th>Values for configuring communication channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Funds Derivation</td>
<td>• AR_Derivation_Send&lt;br&gt;• urn:ariba.com:xi:OnDemand:Derivation&lt;br&gt;• ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com</td>
<td>• CCT_RequestFMDerive_Sender&lt;br&gt;• CCT_InvFMDerive_Sender&lt;br&gt;• urn:ariba.com:xi:OnDemand:Derivation&lt;br&gt;• ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com</td>
<td>• urn:ariba.com:xi:OnDemand:Derivation&lt;br&gt;• MIOut_Sync_WSDL_RequestDerive&lt;br&gt;• MIOut_Sync_WSDL_InvoiceDerive</td>
</tr>
</tbody>
</table>
4. Choose the **Store Certificate** option.
5. Click **Generate**. The system creates two entries, one for the private key and the other with appended `-cert` as suffix as the certificate (public key).
6. Choose the certificate. Under the **Entry** section, click **Export**.
7. In the **Store Key Entry** screen, from the **Files of Type** field, choose **x.509 certificate (*.crt)** file type.
8. In the **File Name** field, enter an appropriate name and click **OK**.

**How to validate messages using the Ariba Procurement Solution**

**Context**

To validate messages using the Ariba Procurement Solution, you need to import the public key exported from Ariba Procurement Solution keystore. You must have copied the certificate into the SAP Process Integration server.

**Procedure**

1. Choose the view where the certificate is stored.
2. In the **Runtime** tab, under the **Entry** section, click **Load**.
3. Choose the required certificate and choose **OK**.

**Configuration for inbound messages received by SAP Process Integration**

The following procedures enable SAP Process Integration to verify messages sent from the Ariba Procurement Solution:

- **How to configure the communication channel for inbound messages received by SAP Process Integration** [page 47]
- **How to configure the sender agreement for inbound messages received by SAP Process Integration** [page 47]
How to configure the communication channel for inbound messages received by SAP Process Integration

Procedure

1. From the SAP Process Integration main page, choose Integration Builder: Configuration, and then log in to the Integration Builder: Configuration application.
2. Choose the sender communication channel for the required integration scenario for which you want to configure the WSS signature.
3. Change to edit mode.
4. Under the Security Parameters section, choose the Select Security Profile option.
5. For the Security Profile field, choose the Web Services Security from the dropdown.
6. In the Persist Duration field, enter 30.
7. Click Save.

Results

The system may display an error because the Sender agreement is not configured.

How to configure the sender agreement for inbound messages received by SAP Process Integration

Procedure

1. From the SAP Process Integration main page, choose Integration Builder: Configuration, and then log in to the Integration Builder: Configuration application.
2. Open the sender agreement for the required communication channel.
3. Change to edit mode.
4. Under Security Settings, specify values for the following fields:
   - Security Procedure - Validate
   - Partner certificate to verify signature (WS)
     - Issuer - Choose the certificate that you imported during the generation of keys
     - Subject - Choose the certificate that you imported during the generation of keys
   - Certification Authority (WS)
     - Keystore View - Choose the private keycertificate that you imported during the generation of keys
5. Click Save.
Next Steps

Activate the communication channel and sender agreement to ensure that your changes are validated.
Configuring parameters

You can change the way Ariba RFCs function by modifying parameters in the RFCs. This section explains how to change \texttt{/ARBA/TVARV} RFC parameters. The material in this section is appropriate for SAP developers or administrators. The following topics explain parameter configuration:

- About \texttt{/ARBA/TVARV} [page 49]
- Parameters [page 50]
- Tables for taxes and charges [page 53]
- Parameters description [page 55]
- Maintaining a temporary directory in \texttt{/ARBA/TVARV} [page 69]

About \texttt{/ARBA/TVARV}

You configure all Ariba RFC parameters through the \texttt{/ARBA/TVARV} table. By editing the parameter configurations in \texttt{/ARBA/TVARV}, you:

- Perform static filtering on data pulls through \texttt{TVARV}-like selection options.
- Configure parameters that affect how Ariba RFCs function.

For information on performing static filtering, see Filtering data [page 133].

For a list of the RFCs affected by parameters in \texttt{/ARBA/TVARV}, see \texttt{/ARBA/PO_IMPORT Package} [page 294].

\texttt{/ARBA/TVARV} contains a special field, called \texttt{TYPE}, which determines whether an entry is for static filtering or parameter configuration. To change parameters, you enter a “P” in the \texttt{TYPE} field. (If you set the field to \texttt{S}, you use \texttt{/ARBA/TVARV} for static filtering; see Filtering data [page 133].)

In your default configuration, \texttt{/ARBA/TVARV} does not contain any parameters. Ariba RFCs that call \texttt{/ARBA/TVARV} already contain certain default values that they use. Enter a value in \texttt{/ARBA/TVARV} only if you want to change the default values.

How to edit \texttt{/ARBA/TVARV} parameters

Procedure

1. In the SAP ERP main screen, navigate to \textbf{System} \rightarrow \textbf{Services} \rightarrow \textbf{Table Maintenance} \rightarrow \textbf{Extended Table Maintenance}.
   
   The Maintain Table Views initial screen appears.
2. Enter /ARBA/TVARV in the Table/View field, and click Maintain.

3. Choose Edit New Entries, or click the New Entries button.

4. Enter appropriate information in the screen that appears. For details, see the following options:
   - Parameters [page 50]
   - Tables for taxes and charges [page 53]

5. To save your changes, choose Table View Save or (press F11).

6. If a transport window appears prompting you to enter the change request, click Create Request to create a new request, or choose an existing request number in the Request field, and then click the check mark to create the transport request.

## Parameters

The following table lists possible /ARBA/TVARV parameters, default values, and the calling functions that reference them. The remaining sections in this section explain what these parameters do and provide other possible values.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Default</th>
<th>Calling Function/Program</th>
</tr>
</thead>
<tbody>
<tr>
<td>BALANCE_SHEET_ACCOUNTS</td>
<td>disallow</td>
<td>/ARBA/GENERAL_LEDGER_EXPORT</td>
</tr>
<tr>
<td>/ARBA/CURR_CONV_BANK_RATE</td>
<td>B</td>
<td>/ARBA/CURRENCY_CONVERSION_EXPORT</td>
</tr>
<tr>
<td>/ARBA/CURR_CONV_DAYS_PAST</td>
<td>60</td>
<td>/ARBA/CURRENCY_CONVERSION_EXPORT</td>
</tr>
<tr>
<td>/ARBA/CURR_CONV_EURO_RATE</td>
<td>EURO</td>
<td>/ARBA/CURRENCY_CONVERSION_EXPORT</td>
</tr>
<tr>
<td>/ARBA/INTERNAL_ORDER_AUART</td>
<td>blank</td>
<td>/ARBA/INTERNAL_ORDER_EXPORT</td>
</tr>
<tr>
<td>/ARBA/INVOICE_ALLOW_DUPS</td>
<td>blank</td>
<td>/ARBA/CR_INVOICE_CREATE/ARBA/CR_ER_INVOICE_CREATE</td>
</tr>
<tr>
<td>/ARBA/LANGUAGE</td>
<td>E</td>
<td>/ARBA/ACCOUNT_CATEGORY_EXPORT/ARBA/MATERIAL_GROUP_EXPORT/ARBA/COST_CENTER_EXPORT/ARBA/GENERAL_LEDGER_EXPORT/ARBA/MINORITY_VENDOR_EXPORT/ARBA/TAX_CODE_EXPORT</td>
</tr>
<tr>
<td>/ARBA/VENDOR_ONLY_COUNTPORGs</td>
<td>1</td>
<td>/ARBA/VENDOR_ONLY_EXPORT</td>
</tr>
<tr>
<td>/ARBA/ER_DOC_TYPE</td>
<td>ER</td>
<td>/ARBA/CR_ER_INVOICE_CREATE</td>
</tr>
<tr>
<td>/ARBA/INVOICE_DOC_TYPE</td>
<td>RE</td>
<td>/ARBA/CR_INVOICE_CREATE</td>
</tr>
<tr>
<td>/ARBA/USE_FI_POSTING_NONPO</td>
<td>blank</td>
<td>/ARBA/CR_INVOICE_CREATE</td>
</tr>
</tbody>
</table>

For more information, see Non-PO (FI) invoices [page 62].
<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Default</th>
<th>Calling Function/Program</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/NOPO_INVOICE_FI_DOC_TYPE</td>
<td>KR</td>
<td>/ARBA/CR_INVOICE_CREATE</td>
</tr>
<tr>
<td>For more information, see Non-PO (FI) invoices [page 62].</td>
<td></td>
<td></td>
</tr>
<tr>
<td>/ARBA/ASSET_TR_TYP_FI_NONPO</td>
<td>020</td>
<td>/ARBA/CR_INVOICE_CREATE</td>
</tr>
<tr>
<td>For more information, see Non-PO (FI) invoices [page 62].</td>
<td></td>
<td></td>
</tr>
<tr>
<td>/ARBA/GROSS_AMOUNT_ROUNDED</td>
<td>blank</td>
<td>/ARBA/CR_INVOICE_CREATE</td>
</tr>
<tr>
<td>BAPI_PO_CREATE_DOC_TYPE</td>
<td>AR</td>
<td>/ARBA/BAPI_PO_CREATE</td>
</tr>
<tr>
<td>BAPI_PO_CREATE_ERP_DOC_TYPE</td>
<td>NB</td>
<td>/ARBA/BAPI_PO_CREATE</td>
</tr>
<tr>
<td>ARBA/USE_BAPI_PO_CRT1</td>
<td></td>
<td>/ARBA/BAPI_PO_CREATE</td>
</tr>
<tr>
<td>If you set the value of this parameter to X, /ARBA/BAPI_PO_CREATE calls the new BAPI BAPI_PO_CREATE1 that supports new structures and fields and is rich compared to BAPI_PO_CREATE. If you do not maintain a value for this parameter, /ARBA/BAPI_PO_CREATE calls the existing BAPI BAPI_PO_CREATE to create purchase orders in SAP ERP.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>/ARBA/BAPI_PR_CRT_ALLOW_DUPS</td>
<td>blank</td>
<td>/ARBA/ACC_PURCHASE_REQUI_CHECK</td>
</tr>
<tr>
<td>For more information, see Budget check requisition variables [page 58].</td>
<td></td>
<td></td>
</tr>
<tr>
<td>/ARBA/PURREQ_DOC_TYP</td>
<td>NB</td>
<td>/ARBA/ACC_PURCHASE_REQUI_CHECK</td>
</tr>
<tr>
<td>For more information, see Budget check requisition variables [page 58].</td>
<td></td>
<td></td>
</tr>
<tr>
<td>/ARBA/HELD_PR_NO</td>
<td>D000000001</td>
<td>/ARBA/ACC_PURCHASE_REQUI_CHECK</td>
</tr>
<tr>
<td>For more information, see Budget check requisition variables [page 58].</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parameter Name</td>
<td>Default</td>
<td>Calling Function/Program</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>-----------------------------------</td>
<td>--------------------------------------------------------</td>
</tr>
</tbody>
</table>
| /ARBA/ENABLE_TLC                     | blank                             | • /ARBA/CR_BAPI_PO_Change  
• /ARBA/CR_BAPI_PO_CREATE1          |
| /ARBA/EVENT_APR                      | X                                 | /ARBA/ADV_PAYMENT_POST                                  |
| /ARBA/CR_ADV_PAY_REMIT_EXPORT        | .\APR\REMITTANCE or specify your file download path.  
For more information, see Advance payment parameters [page 55]. | /ARBA/ADV_PAYMENT_REV_POST                             |
| /ARBA/HEADER_TEXT_ID                 | blank                             | /ARBA/BAPI_PO_CREATE1                                   |
| /ARBA/ITEM_TEXT_ID                   | blank                             | /ARBA/BAPI_PO_CREATE1                                   |
| /ARBA/DMS_STORAGE_CATEGORY           | blank                             | /ARBA/BAPI_PO_CREATE1                                   |
| /ARBA/PO_ATT_ERR_MAIL                | blank                             | /ARBA/BAPI_INVOICE_CREATE                               |
| /ARBA/INV_ATT_ERR_EMAIL              | blank                             | For more information, see Attachments failure email-invoices [page 56]. |

For more information, see Taxes and charges [page 64].

For more information, see Advance payment parameters [page 55].

For more information, see Advance payment parameters [page 55].

For more information, see Purchase order comments [page 63].

For more information, see Purchase order comments [page 63].

For more information, see DMS attachments storage [page 59].

For more information, see Attachments failure email-purchase orders and change purchase orders [page 57].
Related Information

How to edit /ARBA/TVARV parameters [page 49]

Tables for taxes and charges

Following tables support taxes and charges on requisitions, purchase orders, and invoices in SAP ERP.

/ARBA/TAX_CHECK

The /ARBA/TAX_CHECK table is required for requisitions and change requisitions. It has two fields and specifies the company code for which the non-deductible tax amount (tax on item or tax on charges) is considered while reserving the budget for requisitions under a particular company code.

The following table shows sample mappings for the /ARBA/TAX_CHECK table:

Table 4: Sample Mappings in the /ARBA/TAX_CHECK Table

<table>
<thead>
<tr>
<th>Company Code</th>
<th>Field</th>
<th>Is Tax Applied Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>3000</td>
<td></td>
<td>Check to choose applicable tax for the corresponding company code.</td>
</tr>
</tbody>
</table>

Note

If you do not check, the tax amount is ignored while reserving budget for the requisition pertaining to the relevant company code.

/ARBA/COND_MAP

The /ARBA/COND_MAP condition map table is required for purchase orders and change purchase orders. It has three fields and maintains mapping between condition types in Ariba Procurement Solution to relevant condition types in SAP ERP.

For example, the condition type might be SHIPPING in Ariba Procurement Solution and FRB1 in SAP ERP. Buyers must maintain mapping of taxes, charges, and discounts between these condition types in the /ARBA/COND_MAP table.
The following table shows sample mappings for the `/ARBA/COND_MAP` table:

Table 5: Sample Mappings in the `/ARBA/COND_MAP` Table

<table>
<thead>
<tr>
<th>LINETYPECATEGORY</th>
<th>Field For Ariba Procurement Solution Condition Type</th>
<th>LINE_TYPE</th>
<th>Field For Ariba Procurement Solution Condition Type</th>
<th>KSCHL</th>
<th>Field For SAP ERP Condition Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td></td>
<td>SPECIALCHARGE</td>
<td></td>
<td></td>
<td>ZS00</td>
</tr>
<tr>
<td>32</td>
<td></td>
<td>DISCOUNT</td>
<td></td>
<td></td>
<td>RB00</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>FREIGHTCHARGE</td>
<td></td>
<td></td>
<td>FRB1</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>HANDLINGCHARGE</td>
<td></td>
<td></td>
<td>ZB00</td>
</tr>
</tbody>
</table>

**Note**

- You cannot change the `LINETYPECATEGORY` for a particular `LINE_TYPE` as it originates from the Ariba Procurement Solution and is closely integrated with SAP ERP. To add custom charges in Ariba Procurement Solution you can only use the `LINETYPECATEGORY` 16.
- You must maintain the pricing condition type in all the pricing procedures that you want to use in SAP ERP for integration with the Ariba Procurement Solution. You must choose condition types with absolute amount only, that is, condition types with the suffix (value).

`/ARBA/WHT_TYPE`

The `/ARBA/WHT_TYPE` table is required for invoices. It has two fields and maintains mapping between withholding tax type in Ariba Procurement Solution to the relevant withholding tax code in SAP ERP.

The following table shows sample mappings for the `/ARBA/WHT_TYPE` table:

Table 6: Sample Mappings in the `/ARBA/WHT_TYPE` Table

<table>
<thead>
<tr>
<th>Withholding Tax</th>
<th>Field For Ariba Procurement Solution</th>
<th>Withholding Tax</th>
<th>Field For SAP ERP</th>
</tr>
</thead>
<tbody>
<tr>
<td>WTHTAX</td>
<td></td>
<td>T1</td>
<td></td>
</tr>
</tbody>
</table>

Related Information

- How to edit `/ARBA/TVARV` parameters [page 49]
- Integrating taxes and charges [page 227]
Parameters description

This section groups categories of parameters together and explains how to use them in the following topics:

- Advance payment parameters [page 55]
- Allow duplicates for purchase orders [page 56]
- Allow duplicates for invoice [page 56]
- Attachments failure email-invoices [page 56]
- Attachments failure email-purchase orders and change purchase orders [page 57]
- Budget check requisition variables [page 58]
- Currency conversion rates [page 58]
- Days past [page 59]
- DMS attachments storage [page 59]
- Document type for creating expense report invoices [page 59]
- Document type for creating OK-to-Pay invoices [page 60]
- Document type for creating purchase orders [page 60]
- Internal order types [page 61]
- Language for texts [page 61]
- Non-PO (FI) invoices [page 62]
- Purchase order comments [page 63]
- Rounding of gross amount for invoices [page 63]
- Taxes and charges [page 64]
- Using general ledger balance sheet accounts [page 64]
- Vendor export task [page 65]

Advance payment parameters

To support advance payments in SAP ERP, you must set the variables for the following parameters in the /ARBA/TVARV table:

- Parameter: /ARBA/EVENT_APR is required to get data for advance payment request from Ariba Procure-to-Pay to SAP ERP. To fetch the advance payment request data, you must set the variable for Selection value to X. If you keep the variable for Selection value blank, the program calls the old Export Payment Requests event.

Maintain the following values:

- Variable Name: /ARBA/EVENT_APR
- Selection cat.: P
- Number: 0
● **Selection value:** X

- **Parameter:** /ARBA/CR_ADV_PAY_REMIT_EXPORT is required to support advance payment remittance from SAP ERP using direct connectivity.
  - Maintain the following values:
    - **Variable Name:** /ARBA/CR_ADV_PAY_REMIT_EXPORT
    - **Selection cat.:** P
    - **Number:** 0
    - **Selection value:** .\APR\REMITTANCE or specify your file download path.

### Related Information

- Parameters [page 50]
- Integrating advance payments [page 230]

### Allow duplicates for purchase orders

By default, Ariba Procurement Solution does not create duplicate purchase orders. When the integration pushes an approved requisition to SAP ERP, it checks to see if it has already created a purchase order with the same approved Ariba Requisition ID. If so, SAP ERP does not create a new purchase order but returns the existing purchase order number.

If the /ARBA/BAPI_PO_CRT_ALLOW_DUPS parameter is set, the program does not check for duplicates.

### Allow duplicates for invoice

By default, Ariba Procurement Solution does not create duplicate invoices. When the integration pushes an invoice or expense report to SAP ERP, it checks to see if it has already created an invoice with the same reference document for the given supplier and fiscal year. If so, it does not create a new invoice but returns the existing invoice number.

If the /ARBA/INVOICE_ALLOW_DUPS parameter is set, the program does not check for duplicates.

### Attachments failure email-invoices

To support error notification email from SAP ERP when invoice attachments fail in the Ariba Procurement Solution, you must set the variables for the following parameter in the /ARBA/TVARV table:

- **Parameter:** /ARBA/INV_ATT_ERR_EMAIL
Maintain the following values:

- **Variable Name**: /ARBA/INV_ATT_ERR_EMAIL
- **Selection cat.**: P
- **Number**: 0 to n
  - The **Number** value specifies the number of email recipients.
- **INCL/EXCL**: I
- **Option**: EQ
- **Selection value**: Email addresses of the recipients

**Related Information**

About attachments with purchase orders and invoices in the Ariba Procurement Solution integrated with SAP ERP [page 237]
Parameters [page 50]

**Attachments failure email-purchase orders and change purchase orders**

To support error notification email from SAP ERP when purchase order attachments fail in the Ariba Procurement Solution, you must set the variables for the following parameter in the /ARBA/TVARV table:

Parameter: /ARBA/PO_ATT_ERR_MAIL

Maintain the following values:

- **Variable Name**: /ARBA/PO_ATT_ERR_MAIL
- **Selection cat.**: P
- **Number**: 0 to n
  - The **Number** value specifies the number of email recipients.
- **INCL/EXCL**: I
- **Option**: EQ
- **Selection value**: Email addresses of the recipients

**Related Information**

About attachments with purchase orders and invoices in the Ariba Procurement Solution integrated with SAP ERP [page 237]
Parameters [page 50]
Budget check requisition variables

To support real-time budget checks on requisitions, you must set the following variables in the /ARBA/TVARV table:

- /ARBA/BAPI_PR_CRT_ALLOW_DUPS
  - Set X value to allow duplicate values.
  - Set “ “ (blank) values if duplicate values are not allowed.
  
  By default, the Ariba Procurement Solution does not create duplicate requisitions. When the integration pushes a requisition to SAP ERP, it checks to find if it has already created a requisition with the same reference document for the given supplier and fiscal year. If so, it does not create a new requisition but returns the existing requisition number.

- /ARBA/PURREQ_DOC_TYP
  
  Specify the document type for the requisition. If you do not specify this variable, Ariba Procurement Solution uses the NB document type to create requisitions.

- /ARBA/HELD_PR_NO:
  - Selection cat.: P
  - Char20: VREALM_40 (always in upper case)
  - Char20: PREALM_40 (always in upper case)
  - INCL/EXCL: I
  - Option: EQ
  - Selection value: 0
  
  Generates an internal requisition number, for example D000000001, and prevents requisition from moving back to the **Composing** state when you submit a requisition containing all line items as held in the Ariba Procurement Solution. If you do not set this parameter and submit a requisition containing all line items as held, the Ariba Procurement Solution displays the error message: **Please enter items first.**

Currency conversion rates

The program /ARBA/CURRENCY_CONVERSION_EXPORT calls the table /ARBA/TVARV to look up specifications for SAP currency conversion rates for either banks or the Euro:

- /ARBA/CURR_CONV_BANK_RATE
- /ARBA/CURR_CONV_EURO_RATE

The defaults are **B** and **Euro**.

**Note**

You can determine how far back the RFC looks to pull historical currency conversion information to deal with expenses from the past through the parameter: /ARBA/CURR_CONV_DAYS_PAST.
Days past

The DAYS_PAST parameter allows you to configure how far back, in days, the /ARBA/CURRENCY_CONVERSION_EXPORT program must go to get the conversion rates.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CURR_CONV_DAYS_PAST</td>
<td>Keeps track of currency conversion rates</td>
<td>60</td>
</tr>
</tbody>
</table>

Make sure that the days past parameters for Ariba Procurement Solution RFCs are values less than or equal to the scheduled events in the SAP integration table.

DMS attachments storage

To support DMS storage of attachments in SAP ERP, you must set the variable for the following parameter in the /ARBA/TVARV table:

Parameter: /ARBA/DMS_STORAGE_CATEGORY

Maintain the following values:
- **Variable Name**: /ARBA/DMS_STORAGE_CATEGORY
- **Selection cat.**: P
- **Number**: 0
- **INCL/EXCL**: I
- **Option**: EQ
- **Selection value**: SAP-SYSTEM

Related Information

About attachments with purchase orders and invoices in the Ariba Procurement Solution integrated with SAP ERP [page 237]
Parameters [page 50]

Document type for creating expense report invoices

To distinguish invoices created by the expense report program /ARBA/CR_ER_INVOICE_CREATE from invoices created by the OK-to-pay invoice /ARBA/CR_INVOICE_CREATE function module, use different document types for expense report and Ok-to-pay invoices.

By default, the Ariba Procurement Solution uses the custom document type ER for expense reports. This document type does not exist in the SAP ERP standard configuration. You need to create this document type. If
you want to use any other document type make an entry in /ARBA/TVARV table for the variable /ARBA/ER_DOC_TYPE with the document type you would like to use.

Document type for creating OK-to-Pay invoices

By default, the Ariba Procurement Solution uses a custom document type RE for creating invoices using the Ariba OK-to-Pay SAP integration function module, /ARBA/CR_INVOICE_CREATE. If you want to use any other document type, make an entry in /ARBA/TVARV table for the variable /ARBA/INVOICE_DOC_TYPE with the document type to be used.

By default, the Ariba Procurement Solution uses the custom document type RE for OK-to-Pay integration. This document type does not exist in the SAP ERP standard configuration. You need create this document type. If you want to use any other document type make an entry in /ARBA/TVARV table for the variable /ARBA/ER_DOC_TYPE with the document type you would like to use.

Document type for creating purchase orders

For creating purchase orders, there are two parameters that you can use to maintain the document type.

**BAPI_PO_CREATE_DOC_TYPE**

If the document type sent from the Ariba Procurement Solution is ERPCC or blank, then specify the document type that you want to use as a value for this parameter in the /ARBA/TVARV table. If you do not specify a document type in the table, then the default value of the document type will be AR.

You can create custom document types for purchase orders in the BAPI_PO_CREATE_DOC_TYPE parameter. To create custom document types, ensure that you enter values in the TVARV generalized for fields section for the following fields:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable Name</td>
<td>BAPI_PO_CREATE_DOC_TYPE</td>
</tr>
<tr>
<td>Field Name</td>
<td>Leave it blank.</td>
</tr>
<tr>
<td>Selection cat.</td>
<td>P</td>
</tr>
<tr>
<td>Number</td>
<td>0 (zero)</td>
</tr>
<tr>
<td>Ariba Variant</td>
<td>The unique variant name for the Ariba Procurement Solution in SAP ERP for purchase orders. For example, VREALM_1.</td>
</tr>
</tbody>
</table>
### Field Name Description

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Ariba Partition</strong></td>
<td>The unique name of your site for the Ariba Procurement Solution in SAP ERP for purchase orders. For example, PREALM_1.</td>
</tr>
<tr>
<td><strong>INCL/EXCL</strong></td>
<td>This field must have the value i.</td>
</tr>
<tr>
<td><strong>Option</strong></td>
<td>This field must have the value eq.</td>
</tr>
<tr>
<td><strong>Selection value</strong></td>
<td>Sets the low value for filtering and settings for the parameter. For example, ZARB.</td>
</tr>
<tr>
<td><strong>Selection value</strong></td>
<td>This field can be left empty.</td>
</tr>
</tbody>
</table>

### BAPI_PO_CREATE_ERP_DOC_TYPE

If the document type sent from the Ariba Procurement Solution is ERP, then specify the document type that you want to use as a value for this parameter in the /ARBA/TVARV table. If you do not specify a document type in the table, then the default value of the document type will be NB.

### Internal order types

Internal orders contain a field called AUART, which designates their order type. The value of AUART determines whether SAP ERP can use the internal order to create a purchase order. The Ariba Procurement Solution RFCs do not create purchase orders using internal orders that have an AUART value specified with the /ARBA/INTERNAL_ORDER_AUART.

If you do not specify a value for AUART, your Ariba RFCs can create purchase orders from any internal order in which the AUART field of the order master data (AUFK):

- Is not equal to a space.
- Is not marked for deletion (LOERZ).
- Is not already ordered or released.

### Language for texts

To get language-dependent descriptions in languages besides English (if you are running other language instances), you can set the /ARBA/LANGUAGE parameter. Subsequently, the descriptions are displayed in the language set in the parameter. If English language descriptions are not maintained in SAP ERP, set the value to the language in which descriptions need to be shown in the Ariba Procurement Solution.
Non-PO (FI) invoices

To support creation of non-PO (FI) invoices in SAP ERP, you must set the variables for the following parameters in the `/ARBA/TVARV` table:

- **Parameter**: `/ARBA/USE_FI_POSTING_NONPO` is required to create non-PO based invoice posting in SAP ERP. Maintain the following values:
  - **Variable Name**: `/ARBA/USE_FI_POSTING_NONPO`
  - **Selection cat.**: P
  - **Number**: 0
  - **Char20**: Maintain your realm
  - **Char20**: Maintain your partition
  - **INCL/EXCL**: I
  - **Option**: EQ
  - **Selection value**: X

- **Parameter**: `/ARBA/NOPO_INVOICE_FI_DOC_TYPE` is required to get the document to post for FI invoices. By default the document type is KR.
  Maintain the following values:
  - **Variable Name**: `/ARBA/NOPO_INVOICE_FI_DOC_TYPE`
  - **Selection cat.**: P
  - **Number**: 0
  - **Char20**: Maintain your realm
  - **Char20**: Maintain your partition
  - **INCL/EXCL**: I
  - **Option**: EQ
  - **Selection value**: KR

- **Parameter**: `/ARBA/ASSET_TR_TYP_FI_NONPO` is required to maintain the asset transaction type if you use asset accounting assignment.
  Maintain the following values:
  - **Variable Name**: `/ARBA/ASSET_TR_TYP_FI_NONPO`
  - **Selection cat.**: P
  - **Number**: 0
  - **Char20**: Maintain your realm
  - **Char20**: Maintain your partition
  - **INCL/EXCL**: I
  - **Option**: EQ
  - **Selection value**: 020

Related Information

Parameters [page 50]
Integrating non-PO (FI) invoices [page 232]
Purchase order comments

To support sending comments with purchase orders in SAP ERP, you must set the variables for the following parameters in the /ARBA/TVARV table:

- **Parameter: /ARBA/HEADER_TEXT_ID** is required to support purchase orders comments at the header level in SAP ERP.
  - Maintain the following values:
    - **Variable Name**: /ARBA/HEADER_TEXT_ID
    - **Selection cat.**: P
    - **Number**: 0
    - **Char20**: VREALM_40
    - **Char20**: PREALM_40
    - **INCL/EXCL**: I
    - **Option**: EQ
    - **Selection value**: F01

- **Parameter: /ARBA/ITEM_TEXT_ID** is required to support purchase orders comments at the line level in SAP ERP.
  - Maintain the following values:
    - **Variable Name**: /ARBA/ITEM_TEXT_ID
    - **Selection cat.**: P
    - **Number**: 0
    - **Char20**: VREALM_40
    - **Char20**: PREALM_40
    - **INCL/EXCL**: I
    - **Option**: EQ
    - **Selection value**: F02

Related Information

About attachments with purchase orders and invoices in the Ariba Procurement Solution integrated with SAP ERP [page 237]
Parameters [page 50]

Rounding of gross amount for invoices

By default, the /ARBA/CR_INVOICE_CREATE program references the GROSS_AMOUNT instead of the GROSS_AMOUNT_ROUNDED. This results in discrepancy in the total amount calculated after rounding the decimals to two places and can result in invoices sent from the Ariba Procurement Solution failing in SAP ERP.

When you maintain the GROSS_AMOUNT_ROUNDED parameter in the /ARBA/TVARV table, the /ARBA/CR_INVOICE_CREATE program maps the GROSS_AMOUNT_ROUNDED to the GROSS_AMOUNT.
Parameter Name: GROSS_AMOUNT_ROUNDED
  - Variable Name: /ARBA/GROSS_AMOUNT_ROUNDED
  - Field Name: Blank
  - Selection cat.: P
  - Number: 0
  - Char20: Blank
  - Char20: Blank
  - INCL/EXCL: I
  - Option: EQ
  - Selection value: X
  - Selection value: Blank

Related Information

Parameters [page 50]

Taxes and charges

The /ARBA/ENABLE_TLC parameter is required to update the SAPPOLineNumber in the Ariba Procurement Solution.

PARAMETER NAME: /ARBA/ENABLE_TLC

To update the SAPPOLineNumber, ensure that you maintain the following values:

- Variable Name: /ARBA/ENABLE_TLC
- Selection cat.: P
- INCL/EXCL: I
- Option: EQ
- Selection value: X

Using general ledger balance sheet accounts

You use general ledger balance sheet accounts for your procurement needs in the Ariba Procurement Solution. By default, /ARBA/GENERAL_LEDGER_EXPORT does not pull accounts for balance sheets, particularly if SAP ERP is configured not to allow this pull, and returns a hard error message when a user attempts to do so.

To enable /ARBA/GENERAL_LEDGER_EXPORT to pull and use balance sheet accounts, you enter the value allow in the BALANCE_SHEET_ACCOUNTS parameter. To instruct the RFC not to use balance sheet accounts, even if balance sheet accounts are permitted on your system, you enter the value disallow instead.
Vendor export task

By default, Ariba RFCs pull only those vendors from SAP ERP that are found associated with at least one purchasing organization.

The parameter /ARBA/VENDOR_ONLY_COUNTPORGS provides a filtering parameter for vendor pulls. Valid settings are 1 or 0. The default setting is 1.

- If the parameter is set to 1, the RFC chooses only vendors assigned to at least one purchasing organization.
- If the parameter is set to 0, the RFC pulls all vendors without regard to any assigned purchasing organizations.

The default setting of 1 pulls a smaller set of data by executing two select statements within the RFC code. The 0 setting pulls more data, which the RFC refines further through a single selection query.

The Vendor Export task downloads information on the following:

- Supplier
- Supplier location
- Payment terms
- Remittance location
- User data

<table>
<thead>
<tr>
<th>Function Module</th>
<th>/ARBA/VENDOR_EXPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSV Files</td>
<td></td>
</tr>
<tr>
<td>SupplierConsolidated.csv</td>
<td></td>
</tr>
<tr>
<td>PurchaseOrgSupplierCombo.csv</td>
<td></td>
</tr>
<tr>
<td>SupplierLocationConsolidated.csv</td>
<td></td>
</tr>
<tr>
<td>RemittanceLocationConsolidated.csv</td>
<td></td>
</tr>
</tbody>
</table>

The supplier location data contains two parameters that you maintain in the /ARBA/TVARV table: /ARBA/SL_VENDOR_ADDRESS and /ARBA/SL_PARTNER_TYPE.

Prerequisites

- You must maintain at least one of these parameters in the /ARBA/TVARV table while downloading information for supplier locations. If you do not maintain one of these parameters, an error occurs.
- PARAMETER NAME: /ARBA/SL_PARTNER_TYPE

To maintain the partner type filter for the supplier location:

- **Variable Name**: /ARBA/SL_PARTNER_TYPE
- **Field Name**: PARVW
- **Selection cat.**: S
- **Number**: 0

In the Change View “TVARV generalized for fields”: Details section, here is an example of how you can maintain the following values:

- **Char20**: VSAP
- **Char20**: PSAP
- **INCL/EXCL**: I
- **Option**: EQ
Selection value: For example, OA. When you specify OA, the Ariba Procurement Solution integrated with SAP ERP downloads all the supplier locations for the supplier. You can maintain the partner type based on the configuration you have maintained for the partner function for a vendor.

Selection value: Blank

PARAMETER NAME: /ARBA/SL_VENDOR_ADDRESS

To maintain the addresses filter for the supplier location:

- **Variable Name**: /ARBA/SL_VENDOR_ADDRESS
- **Field Name**: Blank
- **Selection cat.**: P
- **Number**: 0

Here is an example of how you can maintain values in the TVARV generalized for fields section:

- **Char20**: VSAP
- **Char20**: PSAP
- **INCL/EXCL**: I
- **Option**: EQ
- **Selection value**: X

When you specify X, the Ariba Procurement Solution integrated with SAP ERP considers the supplier as the supplier location and downloads all addresses for the supplier location. You can maintain the partner type based on the configuration you have maintained for the partner function for a vendor.

Selection value: Blank

You can maintain both the parameters in the /ARBA/TVARV table.

Here is how the Ariba Procurement Solution integrated with SAP ERP processes the supplier location. For example, the following table displays the values associated to the ABC Supplier:

<table>
<thead>
<tr>
<th>Supplier</th>
<th>Purchase Organization</th>
<th>Supplier Location 1</th>
<th>Supplier Location 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC Supplier</td>
<td>1000</td>
<td>1004</td>
<td>1005</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>1006</td>
<td>1007</td>
</tr>
</tbody>
</table>

**Case 1:** When you export the supplier location data, the Ariba Procurement Solution integrated with SAP ERP displays the Supplier Location for ABC supplier as 1004, 1005, 1006 and 1007 in the downloaded CSV file.

**Case 2:** When ABC Supplier is blocked for Purchase Organization 1000, the Ariba Procurement Solution integrated with SAP ERP only displays the Supplier Location as 1006 and 1007 in the downloaded CSV file.

**Note**

- The Ariba Procurement Solution integrated with SAP ERP does not support a supplier location linked to multiple vendors.
- When you do not maintain filters for /ARBA/SL_VENDOR_ADDRESS and /ARBA/SL_PARTNER_TYPE, the Ariba Procurement Solution integrated with SAP ERP displays an error stating that you have not maintained filters for the Supplier Location.
Remittance location data

<table>
<thead>
<tr>
<th>Function Module</th>
<th>/ARBA/REMITTANCE_LOCATION_EXPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSV File</td>
<td>RemittanceLocationConsolidatedExport.csv</td>
</tr>
</tbody>
</table>

You can maintain filters while downloading remittance locations by specifying the following parameters in the /ARBA/TVARV table:

- You can maintain both the /ARBA/RL_VENDOR_ADDRESS and /ARBA/RL_PARTNER_TYPE parameters. However, you must maintain at least one of the parameters while downloading the remittance information.
- You can download supplier as remittance location.
- When you want to download the supplier as the remittance location, you must maintain the /ARBA/RL_VENDOR_ADDRESS filter.

**PARAMETER NAME: /ARBA/RL_PARTNER_TYPE**

To maintain the remittance partner type filter, ensure that you maintain the following values:

- **Variable Name**: /ARBA/RL_PARTNER_TYPE
- **Field Name**: PARVW
- **Selection cat.**: S
- **Number**: 0

Here is an example of how you can maintain values in the **TVARV generalized for fields** section:

- **Char20**: VSAP
- **Char20**: PSAP
- **INCL/EXCL**: I
- **Option**: EQ
- **Selection value**: IP

When you specify IP as the selection value, the Ariba Procurement Solution integrated with SAP ERP considers the Remittance Location (LIFN2) with the Supplier Location (LIFNR) from the WYT3 table and downloads all the address information for the remittance location.

You can maintain the partner type based on the configuration you have maintained for the partner function for a vendor.

**PARAMETER NAME: /ARBA/RL_VENDOR_ADDRESS**

To maintain the remittance address filter, ensure that you maintain the following values:

- **Variable Name**: /ARBA/RL_VENDOR_ADDRESS
- **Field Name**: Blank
- **Selection cat.**: P
- **Number**: 0

Here is an example of how you can maintain values in the **TVARV generalized for fields** section:

- **Char20**: VSAP
- **Char20**: PSAP
- **INCL/EXCL**: I
- **Option**: EQ
- **Selection value**: X

When you specify X as the selection value, the Ariba Procurement Solution integrated with SAP ERP considers the supplier as the remittance location and downloads all the address details for the supplier.
You can maintain the partner type based on the configuration you have maintained for the partner function for a vendor.

You can have the following combinations for the supplier location and remittances location:

<table>
<thead>
<tr>
<th>Supplier Location Address /ARBA/SL_VENDOR_ADDRESS</th>
<th>Supplier Location Partner Type /ARBA/SL_PARTNER_TYPE</th>
<th>Remittance Location Address /ARBA/RL_VENDOR_ADDRESS</th>
<th>Remittance Location Partner Type /ARBA/RL_PARTNER_TYPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintain</td>
<td>-</td>
<td>Maintain</td>
<td>-</td>
</tr>
<tr>
<td>Maintain</td>
<td>-</td>
<td>-</td>
<td>Maintain</td>
</tr>
<tr>
<td>Maintain</td>
<td>-</td>
<td>Maintain</td>
<td>Maintain</td>
</tr>
<tr>
<td>Maintain</td>
<td>Maintain</td>
<td>Maintain</td>
<td>-</td>
</tr>
<tr>
<td>Maintain</td>
<td>Maintain</td>
<td>-</td>
<td>Maintain</td>
</tr>
<tr>
<td>Maintain</td>
<td>Maintain</td>
<td>Maintain</td>
<td>Maintain</td>
</tr>
</tbody>
</table>

**Note**

- When you maintain only the filter for the supplier location address /ARBA/SL_VENDOR_ADDRESS and do not maintain the supplier location and remittance location for the supplier, the Ariba Procurement Solution integrated with SAP ERP considers the supplier as the supplier location and remittance location.
- When you maintain the supplier location but do not maintain the remittance location, the Ariba Procurement Solution integrated with SAP ERP considers the supplier location as the remittance location.
- When you do not maintain filters for /ARBA/RL_VENDOR_ADDRESS and /ARBA/RL_PARTNER_TYPE, the Ariba Procurement Solution integrated with SAP ERP displays an error stating that you have not maintained filters for the Remittance Location.

**PARAMETER NAME: /ARBA/RL_BANK_DETAILS**

To maintain the bank details filter, ensure that you maintain the following values:

- **Variable Name**: /ARBA/RL_BANK_DETAILS
- **Field Name**: Blank
- **Selection cat.**: P
- **Number**: 0

Here is an example of how you can maintain values in the **TVARV generalized for fields** section:

- **Char20**: VSAP
- **Char20**: PSAP
- **INCL/EXCL**: I
- **Option**: EQ
- **Selection value**: X

The Ariba Procurement Solution integrated with SAP ERP processes remittance locations as follows:

<table>
<thead>
<tr>
<th>Maintain filter for /ARBA/RL_BANK_DETAIL</th>
<th>Downloads Bank Details for Remittance Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintain</td>
<td>Yes</td>
</tr>
<tr>
<td>Do not maintain</td>
<td>No</td>
</tr>
</tbody>
</table>
Here is an example of how the supplier location and remittance location is mapped for a vendor:

Vendor 1050 has 1001 and 1002 as supplier locations. The remittance location associated to Vendor 1050 is 2000. This remittance location is associated to two payment banks - Dresden (10050033) and Hypo (19652993).

When you download the SupplierLocationConsolidated.csv, you can see 1001 and 1002 as the LocationID for VendorID 1050. The RemittanceLocationConsolidated.csv displays the two payment banks for both the 1001 and 1002 supplier locations.

Maintaining a temporary directory in /ARBA/TVARV

How to maintain a temporary directory while running master data directly [page 69]
How to maintain a temporary directory while running transaction data directly [page 70]

How to maintain a temporary directory while running master data directly

Procedure

1. In the Transaction code /osm 30, go to the /ARBA/TVARV table.
2. Enter the following values:
   ○ Variable Name: /ARBA/TEMP_DIRECTORY
   ○ Fieldname: Blank
   ○ Selection category: P
   ○ Number: 0
   In the TVARV generalized for fields section, specify the following:
   ○ Char20: Blank
   ○ Char20: Blank
   ○ INCL/EXCL: Blank
   ○ Option: Blank
   ○ Selection value: /ARBA/TEMP_DIRECTORY
   ○ Selection value: To be specified by you.

Related Information

How to install client certificates [page 244]
How to maintain a temporary directory while running transaction data directly

Procedure

1. In the Transaction code /osm 30, go to the /ARBA/TVARV table.
2. Enter the following values:
   - **Variable Name**: /ARBA/TEMP_DIRECTORY - For purchase order header status and remittances.
   - **Variable Name**: /ARBA/APR_CREATE - For advance payment request creation.
   - **Variable Name**: /ARBA/APR_CANCEL - For advance payment request cancellation.
   - **Fieldname**: Blank
   - **Selection category**: P
   - **Number**: 0
     - In the TVARV generalized for fields section, specify the following:
       - **Char20**: Blank
       - **Char20**: Blank
       - **INCL/EXCL**: Blank
       - **Option**: Blank
       - **Selection value**: /ARBA/TEMP_DIRECTORY
       - **Selection value**: To be specified by you.

Related Information

- Integrating transactional data directly using the user interface [page 243]
- Integrating advance payments [page 230]
Configuring SAP ERP

About configuring time-out intervals [page 71]
Verifying data [page 72]
How to deactivate the company code validation [page 74]
Configuring service sheets [page 74]

About configuring time-out intervals

SAP ERP has a time-out limit on RFC transactions. The limit makes sure no resources stay locked in the SAP ERP inadvertently. The SAP ERP time-out interval is defined by the profile parameter rdisp/max_wprun_time. The default value is 300 seconds.

If you have a large purchase order to export (with a large number of line items) or a large amount of data to import, you might encounter this time-out limit. In this case, the SAP ERP could cancel the job. To avoid this problem, you change your SAP ERP configuration to lengthen the time-out interval. Change the interval on your central instance as well as any application servers that communicate with the Ariba Procurement Solution. For the change to take effect, restart each SAP ERP server.

Note
Make sure to enlist the help of your BASIS administrator to make these changes.

How to set the time-out interval

Context

You can import or export a large amount of data, but you might encounter a time-out limit. To avoid this issue, you change your SAP ERP configuration to increase the time-out interval.

Procedure

1. Log in as the SAP Administrator.
2. Run the transaction code RZ10.
3. Enter the Instance Profile.
4. Enter the most current version (you create a new version based on the most current one).
5. In the Edit profile panel, click the **Extended Maintenance** radio button.
6. Click **Change**.
7. Scroll down to **Parameter** name rdisp/max_wprun_time.
   
   If the value does not appear in your profile, create it by choosing **Parameter** ➤ **Change**.
8. Create or change the value of rdisp/max_wprun_time to something longer, for example, 3000 (for 3000 seconds).
9. Choose **Parameter** ➤ **Change**.
10. Click **Copy**.  
    This step makes a copy of the new profile. The status line displays the message *Changes were made.*
11. Navigate back to the initial **Edit Profiles** screen, and save the changes.
12. Choose **Profile** ➤ **Activate**.
13. Restart your SAP server for the profile changes to take effect.
14. Repeat the steps in this procedure for each application server that exchanges data with the Ariba Procurement Solution.

---

### Verifying data

In some cases, the Ariba Procurement Solution does not pull data from SAP ERP unless that data has been set up in a particular manner in SAP ERP. This section describes a few places to check the data on your SAP ERP, to make sure it has been set up to match the requirements of the Ariba Procurement Solution.

If an integration entry fails, check this section again to look for problems.

- About the bank selling rate [page 72]
- Blank descriptions [page 73]
- Vendor restrictions [page 73]
- Vendor ERS tax status [page 73]
- Catalog item validation [page 74]

---

### About the bank selling rate

The Ariba Procurement Solution imports currency conversion information from SAP ERP to make sure all purchase orders from Ariba Procurement Solution use currencies valid in SAP ERP. In the default configuration, the bank selling rate is B. The currency conversion pull imports all currency conversions associated with the bank setting rate B. The import task fails if the rate is not configured in the system. If you want to specify a different bank selling rate, add an entry in the /ARBA/TVARV table for the **BANK_SELLING_RATE_SYMBOL** parameter.
How to find the bank selling rate symbol

Procedure

1. Run transaction spro.
2. Choose General Settings > Currencies > Check Exchange Rate Types. Set the symbol, if necessary.

Blank descriptions

For most data in SAP ERP, such as Purchasing Groups, the SAP Process Integration event imports (and uses) both ID and description fields from SAP ERP. If either value is blank, the import task ignores that item, and it does not appear in the Ariba Procurement Solution.

Check your SAP ERP to make sure all material groups and purchasing groups have valid—that is, non-blank—ID and description fields.

Vendor restrictions

The Ariba Procurement Solution integration with SAP ERP has some restrictions on vendors. To import all available vendors, make sure none of the vendors have partner functions specifying multiple ordering addresses. The SAP ERP integration imports one ordering address only, so users cannot choose from among multiple ordering addresses.

The Ariba Procurement Solution does not import vendors that are blocked. SAP ERP uses special codes to designate blocked vendors. The field SPERQ in table LFA1 specifies the code that SAP ERP uses for blocked vendors. The value you need to check for the SPERQ field is in the table TQ04A in the field SPERRFKT.

Vendor ERS tax status

The Ariba Procurement Solution supports vendors from SAP ERP that offer evaluated receipt settlement (ERS) agreements. These agreements have different tax rates depending on many factors, such as the tax laws in the location of the supplier and requester.

Purchase orders from the Ariba Procurement Solution for SAP ERP contain a field to determine whether a vendor has ERS tax information. The Ariba Procurement Solution always sets a value for the ERS tax field. If the Ariba Procurement Solution has no tax ERS information about a vendor, it sets this field to a blank, empty, or null value.

Some SAP ERP systems require you to set a default value for empty tax code values. Such systems return an error message during purchase order pushes if you have not configured them to handle empty values.

- To correct this problem, configure SAP ERP to set a default tax code.
- If you cannot configure a default tax code, change the message type from error to information or warning in SAP ERP.
Catalog item validation

Catalog items are validated against the purchasing organization. The vendor for each catalog item must be defined in the purchasing organization.

If a user picks a catalog item that originated in SAP ERP, the Ariba Procurement Solution validates the vendor of that item against the user’s purchasing organization. If the vendor is not defined in the purchasing organization, the Ariba Procurement Solution flags the field as invalid and does not allow the user to submit the purchase requisition until the field has been corrected.

How to deactivate the company code validation

Context

You can avoid the dump error when creating receipts from MIGO and invoices from MIRO by deactivating the validation at the company-code level.

Procedure

1. In SAP ERP, run the transaction code OBBH.
   This transaction code validates the fields in the table BSEG. For example, the fields in the table BSEG validates ZZREGION at the company-code level.
2. Deactivate the validation at the company-code level.

Configuring service sheets

To enable service sheet response in SAP ERP, perform the following procedures:

   - How to create an output type for the entry sheet application [page 75]
   - How to maintain condition records for outbound transactions: service sheets [page 75]

Related Information

About integration of service procurement in the Ariba Procurement Solution integrated with SAP ERP [page 234]
How to create an output type for the entry sheet application

Procedure

1. Start transaction NACE.
2. Select application ES Entry Sheet and click Output types.
3. In the Processing routines section, create an entry as follows:
   ○ Transmission Medium - 8 (Special Function)
   ○ Program - /ARBA/ENTRYSHEET_UPDATE
   ○ FORM routine - SEND_ENTRYSHEET_RESPONSE
   ○ Save the entry.
4. In the Partner Functions section, create an entry as follows:
   ○ Medium - 8 (Special Function)
   ○ Function - VN
   ○ Description - Vendor
   ○ Save the entry.

Related Information

About integration of service procurement in the Ariba Procurement Solution integrated with SAP ERP [page 234]
How to maintain condition records for outbound transactions: service sheets [page 75]

How to maintain condition records for outbound transactions: service sheets

Procedure

1. In the main screen when you start the transaction NACE, select application ES Entry Sheet and click Condition records.
2. Click the relevant output type.
3. Click Condition records.
4. Select Purchasing Output Determination: Document Type option.
5. Click the Execute button and maintain the following parameters by creating a new entry:
   ○ Document Type - NB (for example)
   ○ Partner Function - VN
   ○ Partner Number - 1000 (for example)
6. Save the entry.

**Related Information**

- About integration of service procurement in the Ariba Procurement Solution integrated with SAP ERP [page 234]
- How to create an output type for the entry sheet application [page 75]
Verifying connection information [page 77]
Receiving tolerances [page 77]
How to enable generation of an ERP order by default [page 78]
Specifying the preferred ordering method [page 79]
Specifying a unique number for purchase orders [page 79]
Enabling cancel order integration [page 79]
How to enable creation of split orders based on currency [page 80]
Specifying ad-hoc ship-to addresses to requisition and purchase orders [page 82]

Verifying connection information

In web-services based integration, for all export tasks, the URL of the SAP Process Integration is configured through the Ariba Administrator.

A generic URL looks similar to the following:

SoapURL="http://XISOAPAdapter/MessageServlet?channel=:<BusinessSystemName>:<Communication ChannelName>";

For example:


where,

- sapnw01: is the server name
- 50000: is the port
- BS_Procurement: is the business system name and
- CC_ChangePurchaseOrder_WSDL_Sender: is the communication channel name

Receiving tolerances

The Ariba Procurement Solution allows users to enter receiving information about the items they order. To support this feature, the Ariba Procurement Solution and SAP ERP both need to receive items in the same way. For example, both must have similar receiving tolerances, or similar ways of allowing users to account for times when they receive fewer or more items than they have ordered.
Receiving system parameters

The Ariba Procurement Solution allows you to configure receiving tolerances through the following settings in the system parameters table:

- Application.Procure.OverReceivingOperator
- Application.Procure.OverReceivingPercentage
- Application.Procure.OverReceivingQuantity
- Application.Procure.OverReceivingValue
- Application.Procure.UnderReceivingOperator
- Application.Procure.UnderReceivingQuantity
- Application.Procure.UnderReceivingValue
- Application.Procure.UnderReceivingPercentage

These values have to be set from the Service Manager. To set values for these parameters, please have your Designated Support Contact log a service request and an Ariba Customer Support representative will contact you.

Make sure you know the following facts about Ariba Procurement Solution receiving integration:

- Ariba Procurement Solution exports both UnderReceiving and OverReceiving Percentages when exporting a purchase order.
- Ariba Procurement Solution does not import tolerances from SAP ERP, but instead uses only those tolerances in the system parameters table.
- For receiving integration to work, make sure the receiving tolerances in the Ariba Procurement Solution match the receiving tolerances in SAP ERP.

How to enable generation of an ERP order by default

Context

When you create a requisition for the first time, a direct order or PCardOrder is generated on approval. To ensure that an ERP Order is generated by default, you must change the default behavior.

Procedure

1. Log in to the Service Manager as Superuser.
2. From the navigation bar on the left, under Site Manager, click Customer Sites.
3. Log in as Customer Support Admin to the relevant realm.
4. Click Customization Manager.
5. Click Advanced on the top right corner.
6. Click Parameters.
7. Under Search Filters, search for ordermethods.
8. Click Edit.
9. Move ariba.sap.server.SAPOERP to the top of the list.
10. Click OK.
11. Click Save.

Specifying the preferred ordering method

The preferred ordering method determines the format in which the purchase order must be sent to the supplier based on the supplier location. The purchase order can be sent in the following formats:

- Fax
- URL
- Print
- cXML

By default the value is set to URL. To ensure that the purchase order is downloaded in a CSV file, you must set the value of the preferred ordering method to Print. To set your preferred ordering method, please have your Designated Support Contact log a service request and an Ariba Customer Support representative will contact you.

Specifying a unique number for purchase orders

By default, the order numbers in the Ariba Procurement Solution are generated starting from EP10. To specify your preferred unique numbers for purchase orders, please have your Designated Support Contact log a service request and an Ariba Customer Support representative will contact you.

Enabling cancel order integration

To enable cancel order integration, the parameter Application.Procure.UseCancelOrderIntegration must be set to Yes. To set the value for this parameter, please have your Designated Support Contact log a service request and an Ariba Customer Support representative will contact you.
How to enable creation of split orders based on currency

Context

If you create a requisition that has line items having different currencies, then the order that will be created must be split based on the different currencies. This is because SAP ERP does not accept a purchase order that contains line items having different currencies.

Procedure

1. Log in to the Service Manager as Superuser.
2. From the navigation bar on the left, under Site Manager, click Customer Sites.
3. Log in as Customer Support Admin to the relevant realm.
4. Click Customization Manager.
5. Click Field Configuration.
6. In Browse Classes, enter procurelineitem and click Search.
7. Click ProcureLineItem.
8. Click Create Field on the bottom right side and choose Computed Field.
9. In the Label field, enter a label for the field.
10. In the Expression field, click Add.
11. Expand Amount and click Currency.
12. Click Add next to the currency field.
13. Click on the Advanced tab
14. Choose Split Order on this Field and click OK.
15. Click Customizing on the top of the page.

About ERP orders

ERP orders are purchase orders that the Ariba Procurement Solution pushes to an ERP (such as SAP ERP) instead of sending them to the vendor. The Ariba Procurement Solution expects ERP to handle the ERP orders and transmit them to vendors.

Be careful about receiving integration with ERP orders because SAP ERP users and processes can change an ERP order after the ERP order is in SAP ERP. Such changes can cause errors, especially if SAP ERP applies release strategies to purchase orders from Ariba Procurement Solution. SAP ERP does not allow you to receive against a purchase order until all the release strategies have been fulfilled.
See your Ariba Solutions Delivery representative to obtain more information and to decide if ERP orders are appropriate for your implementation.

**Support for split amount charges**

Users using the Ariba Procurement Solution integrated with SAP ERP can send accounting information for a line item split by amount in a purchase order, change purchase order, and invoice to an SAP ERP using the file-based or web services-based integration. SAP ERP buyers can also send the accounting information to suppliers on Ariba Network.

**Requirements for split by amount accounting**

You require the following to send split by amount accounting information:

- The Ariba Procurement Solution users and SAP ERP buyers must apply the SAP Note [1387485]: *Account Assignment Distribution Value changes on Replication* for split by amount to use this functionality on their SAP ERP. You must also activate the MM Multiple Account Assignment business function (*LOG_MM_MAA_1*) to use amount-based distribution for amount and percentage distribution. This function is available as part of the SAP Enhancement Package 4 for SAP ERP 6.0.

  **Note**

  When you activate the MM Multiple Account Assignment business function (*LOG_MM_MAA_1*), it automatically also enables the valuated goods receipt for document items with multiple account assignments. The Ariba Procurement Solution integrated with SAP ERP does not support receipt that contain the accounting information. However, you can customize the goods receipt transaction to support sending accounting information in receipts. For more information, see Customizing the Goods Receipt Transaction to Support Accounting Information [page 82].

- Procurement users must enable the parameter *AllowSAPSplitByAmountInPOExport* to send accounting information split by amount to SAP ERP.

- To support split accounting by amount, changes have been made to the purchase order, change purchase order, and invoice integration events. You must download and import the necessary mapping changes in your SAP Process Integration and transport requests into your SAP ERP.

**Additional References**

- The BAPI /ARBA/BAPI_PO_CREATE1 supports split accounting by amount in purchase orders. The existing /ARBA/BAPI_PO_CREATE is no longer applicable.
- The BAPI /ARBA/BAPI_PO_CHANGE supports split accounting by amount in change purchase orders.
- Mapping changes are available in the WSDL accountassignment amount field.
- The BADI /ARBA/PO_IMPORT-publish_po_create1 supports customization for split accounting by amount in purchase orders.
Customizing the goods receipt transaction to support accounting information

Use the following BADI to customize the goods receipt transaction to support the accounting information in receipts:

**BADI Definition Name**: /ARBA/GR_CREATE

**Method**: GOODS_RECEIPTS_BDCDATA

Ensure that you populate the BDCDATA for the split accounting information in the GOODS_RECEIPTS_BDCDATA method.

Specifying ad-hoc ship-to addresses to requisition and purchase orders

The Ariba Procurement Solution integrated with SAP ERP allows buyers to integrate the ad-hoc Ship-To addresses available in the purchase orders sent from Ariba Procurement Solution.

When a buyer enables the parameter `Application.Procure.AllowAdhocShipToAddress` in their Ariba Procurement Solution, they can create an ad-hoc ship-to address while creating a requisition or editing the purchase order. The Export Purchase Order to ERP task exports the ad-hoc ship-to-address available in the purchase order to SAP ERP. The Ariba Procurement Solution integrated with SAP ERP processes the ad-hoc ship-to-address instead of the address ID stored at the plant-level on SAP ERP. The ad-hoc ship-to-address gives buyers the flexibility to receive the goods at a different location.

The ad-hoc ship-to address is stored in the PurcOrdLineDetails.csv file and exported to the SAP ERP through the Export Purchase Order to ERP task.

Prerequisites

Buyers must enable the parameter `Application.Procure.AllowAdhocShipToAddress` in their Ariba Procurement Solution to create an ad-hoc ship-to address while creating a requisition or editing the purchase order.
Limitations

- When a buyer specifies an ad-hoc ship-to address for line items, the ad-hoc Ship-To address is stored as the delivery address on SAP ERP. When a buyer resets the delivery address to use the plant address when changing purchase orders in the Ariba Procurement Solution, the delivery address is updated to the plant address accordingly on SAP ERP. However, the delivery address does not get updated to the plant address for buyers using SAP ERP 6.0 EHP0. Buyers must ensure that they manually choose the delivery address to use the plant address when changing purchase orders on SAP ERP.
- When a buyer using the Ariba Procurement Solution creates a purchase order having an ad-hoc Ship-To address, the purchase order is created successfully with the ad-hoc Ship-To address on SAP ERP. However, when an invoice is created against this purchase order, the invoice fails with a tax calculation error as the Ariba Procurement Solution cannot identify the tax code for the ad-hoc Ship-To address. To process invoices with ad-hoc Ship-To addresses, buyers must ensure that they specify a valid tax code for the ad-hoc Ship-To address while creating an invoice in the Ariba Procurement Solution.

Customization

Customers can also use the BAdI to customize and add additional fields on a purchase order. The following BAdIs are available for customization:

- **BAdI for /ARBA/BAPI_PO_CREATE1**
  - /ARBA/PO_IMPORT
  - Method PUBLISH_PO_CREATE1
- **BAdI for /ARBA/BAPI_PO_CHANGE**
  - /ARBA/PO_CHANGEORDER
  - Method PUBLISH_PO_CHANGE

Ship-to address fields with ad-hoc ship-to information

Ariba Procurement Solution integrated with SAP ERP processes the following fields for the ad-hoc ship-to address available in the PurchOrdLineDetails.csv file:

<table>
<thead>
<tr>
<th>Field Name</th>
<th>This field stores...</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLANT</td>
<td>The temporary Plant ID for the newly created ad-hoc ship-to address. When the requisition does not have an ad-hoc ship-to address, this field stores the Plant ID for SAP ERP.</td>
</tr>
<tr>
<td>HAS_ADHOC_SHIPTO_ADDRESS</td>
<td>The <strong>Yes</strong> value to denote that the requisition stores an ad-hoc ship-to address.</td>
</tr>
<tr>
<td>CITY</td>
<td>The name of the city.</td>
</tr>
<tr>
<td>POSTAL_CODE</td>
<td>The postal code of the city.</td>
</tr>
<tr>
<td>Field Name</td>
<td>This field stores...</td>
</tr>
<tr>
<td>--------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ADDRESS_LINES</td>
<td>The street information for the ad-hoc ship-to address. Buyers can view the address details for the ad-hoc ship-to address in the <strong>Delivery Address</strong> tab while creating or updating the purchase order in SAP ERP. When the requisition does not store the ad-hoc ship-to address, this field stores the address details of the Plant ID for SAP ERP.</td>
</tr>
<tr>
<td>STATE</td>
<td>The state for the ship-to address.</td>
</tr>
<tr>
<td>COUNTRY</td>
<td>The country for the ship-to address.</td>
</tr>
<tr>
<td>DEFAULT_SHIPTO</td>
<td>The Plant ID for SAP ERP. The Plant ID is derived from the user profile data. Ensure that you do not store the ad-hoc address in your SAP ERP.</td>
</tr>
</tbody>
</table>

Although the following fields are available in the Ariba Procurement Solution system, they are not sent to SAP ERP:

- NAME
- STREET 2
- STREET 3
- PHONE
- FAX
- EMAIL
Integrating master data

About master data [page 85]
Importing master data [page 86]
Importing master data to Federated Process Control-enabled systems [page 102]
Incremental load events [page 113]
Company code import RFC example [page 129]
Customizing RFC [page 130]
Filtering data [page 133]
Master data integration events [page 142]
Integrating master data using the direct connectivity integration method from SAP ERP [page 152]
Integrating master data using the mediated connectivity integration method [page 161]

About master data

Master data consists of general information from SAP ERP that is used in the Ariba procurement solutions to construct business documents such as requisitions, receipts, and invoices.

The Ariba procurement solutions provide the integration events for standard data imports from SAP. Because your SAP ERP implementation might contain additional data to be included with the Ariba procurement solutions purchase orders, invoices, or receipts, you modify these data imports to include such data.

The information presented in this chapter is intended for SAP ERP administrators and developers. Performing the tasks detailed in this chapter requires familiarity with the ABAP/4 (Advanced Business Application Programing) language and proficiency in basic SQL tasks.

This chapter describes how to create and edit objects in your SAP instance. When you make changes to Ariba procurement solutions objects in SAP, you might encounter the following warning: “Only urgent repairs in the consolidation system.” This warning appears when you change objects on both consolidation and development systems and indicates that you are making changes to objects that originated in another system. In this particular case, the objects originated from Ariba. The SAP ERP client expects you to develop your own objects on a development system rather than to change objects from third-party vendors. Therefore, in the context of this chapter, it is safe to disregard this warning message.
Importing master data

This section has the following topics:

- Required Data [page 87]
- Requirements for Importing Master Data [page 88]
- Limitations for Importing Master Data [page 88]
- Backward Compatibility [page 88]
- Simplified Master Data Integration [page 89]
- Defining authorizations for running Ariba Procure-to-Pay transactions [page 90]
- Support Integration for Outline Agreements in SAP [page 98]

Master data consists of general information stored in SAP that is used in Ariba procurement solutions to construct requisitions, receipts, or invoices. Transaction data includes information like remittance advice, invoice, purchase orders, and receipts.

You can import master data using only one of the following methods:

- Using the Data Transfer Tool.
- Running integrations events manually from Ariba Administrator.
- Importing directly without using the Ariba Integration Toolkit through the Direct Connectivity Integration method.
- Integrating master data directly from an SAP ERP system to the Ariba Procurement Solution system seamlessly using the SAP Process Integration through the Mediated Connectivity Integration method.

Buyers using the Ariba procurement solutions integrated with SAP can do the following:

- Perform a simplified master data integration for supplier, supplier location, payment terms, remittance location, and user data between the Ariba procurement solution and SAP ERP.
- Integrate master data through the Direct Connectivity Integration method from an SAP system to an Ariba procurement solution system more efficiently using the interface in the ARBA/MASTER_DATA_EXPORT program.
- Integrate master data directly from SAP ERP to an Ariba procurement solution system seamlessly directly or using the Ariba Integration Toolkit.
- Run the Incremental Load while running the master data events directly or using the Ariba Integration Toolkit.
- Integrate master data from SAP ERP to Ariba procurement solutions in a Federated Process Control-enabled environments. You can use the Direct Connectivity or Mediated Connectivity integration methods to integrate master data with Federated Process Control-enabled systems.
- Pull the Payment Terms Language master data in different languages directly or using the Ariba Integration Toolkit.
- Run the Full and Incremental Delete operation directly through the Direct Connectivity Integration method or using the Ariba Integration Toolkit while integrating master data from the buyer’s SAP ERP.
- Integrate master data directly from an SAP ERP system to the Ariba Procurement Solution system seamlessly using the SAP Process Integration through the Mediated Connectivity Integration method.

Related Information
Required data

Ariba procurement solutions imports only the data it needs to create properly formed purchase orders, invoice, and receipt information for SAP.

Ariba procurement solutions imports the following data from SAP:

- Account assignment categories
- Accounting field display status
- Language-specific names for data imports
- Asset accounts
- Cost centers
- Company Codes
- General ledger accounts
- Internal orders
- WBS (Work Breakdown Structure) elements
- Currency conversion rates
- Material groups
- Plants
- Purchasing organizations
- Purchasing groups
- Vendors
- Tax codes
- Supplier Locations
- Payment terms
- Remittance Locations
- User and User groups

Optionally, you can import the following data from SAP with some configuration on your part:

- Users
- Release authorizations
- User release authority

Note

Vendor master payment terms import is available in Ariba procurement solutions.

The Ariba Procurement Data Import and Administration Guide discusses how to configure this optional data and also about imports and integration events related to SAP.
Requirements for importing master data

- You must download and export the transport requests into your SAP ERP system. Ensure that you implement the following SAP notes before importing the master data to avoid a dump occurring with the runtime error:
  - 1402826 - Supplement to Note 548131
  - 1716777 - Runtime error IMPORT_WRONG_END_POS when displaying class
- Before you import the master data, you must create the Authorization Object class. For more information, see Defining authorizations for running Ariba Procure-to-Pay transactions [page 90].

Note

When you import large amounts of master data from your procurement ERP system, ensure that you run the `/ARBA/MASTER_DATA_EXPORT` task in the background.

Limitations for importing master data

The following limitations are applicable while importing the master data:

- You cannot archive the master data (supplier information, payment terms, remittance location, and user data) that has been exported directly without the Ariba Integration Toolkit.
- Ariba Procurement Solution integrated with SAP does not support a supplier location linked to multiple vendors.
- The legacy supplier export task is no longer supported. You must use the Export Supplier Data (Consolidated File) task.
- You can create payment terms with day limit in SAP. But Ariba Procurement Solution integrated with SAP does not support the day limit. If you have maintained payment terms in your SAP ERP system, Ariba Procurement Solution integrated with SAP downloads only one entry.

For more limitations, see the specific sections for each type of incremental load export tasks in Incremental Load Events [page 113].

Backward compatibility

This feature is compatible with previous versions of the Ariba Cloud Integration releases. However, if you have customization for Customer Exit functions, then it is required that you migrate the changes to the corresponding BADI.
Simplified master data integration

To export the master data from your SAP system to your Ariba Procurement Solution system, a custom table /ARBA/FIELD_MAP must be maintained to map the SAP and Ariba Procurement Solution system field values. Ensure that you maintain the following values:

<table>
<thead>
<tr>
<th>Field Name in the /ARBA/FIELD_MAP</th>
<th>Key</th>
<th>Initial Values</th>
<th>Data Element</th>
<th>Data Type</th>
<th>Length</th>
<th>Decimals</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STRUCTURE</td>
<td>Selected</td>
<td>Selected</td>
<td>/ARBA/STRUCTURE</td>
<td>CHAR</td>
<td>30</td>
<td>0</td>
<td>The structure name used in the TABLES parameter for the RFC function.</td>
</tr>
<tr>
<td>FIELD_SAP</td>
<td>Selected</td>
<td>Selected</td>
<td>/ARBA/FNAME_SAP</td>
<td>CHAR</td>
<td>30</td>
<td>0</td>
<td>The field name in the SAP system.</td>
</tr>
<tr>
<td>FIELD_ARIBA</td>
<td></td>
<td></td>
<td>ARBA/FNAME_ARIBA</td>
<td>CHAR</td>
<td>255</td>
<td>0</td>
<td>The field name in the Ariba system that you want to receive in the output .csv file.</td>
</tr>
</tbody>
</table>

**Note**

If you add fields to the customized RFC structure, add this field and the corresponding column name to the /ARBA/FIELD_MAP table to have a preferred column name in the output csv files. If you do not add this field, Ariba Procurement Solution integrated with SAP displays the SAP field name for that column in the Ariba Procurement Solution system.

Here is an example of how a field name in SAP can be maintained to a corresponding preferred column name in the /ARBA/FIELD_MAP table.

For example,

<table>
<thead>
<tr>
<th>STRUCTURE</th>
<th>Field Name in the Ariba Procurement Solution system</th>
<th>Field Name in the Ariba Procurement Solution system</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/PAYMENT_TERMS</td>
<td>NAME</td>
<td>UniqueName</td>
</tr>
<tr>
<td>/ARBA/PAYMENT_TERMS</td>
<td>PRZTY</td>
<td>DiscountType</td>
</tr>
<tr>
<td>/ARBA/PAYMENT_TERMS</td>
<td>RATNR</td>
<td>InstallmentNumber</td>
</tr>
</tbody>
</table>
Defining authorizations for running Ariba Procure-to-Pay transactions

If you want to authorize only selected users to execute these transactions, you must create the profiles for the authorization objects /ARBA/RFC and /ARBA/PROG. These authorization objects are part of the transport and they must be assigned to a role in PFCG transaction code. If the role exists for the RFC Communication Channel user, assign the authorization object to the role directly, or create a new role and profile, and then assign the authorization object.

In PFCG, while assigning the authorization object to the role, check FUGR for RFC_TYPE and for RFC_NAME, provide the names of the Ariba Procure-to-Pay RFC function modules based on the user’s transaction level authorization. For example, /ARBA/BAPI_PO_CREATE1 and other Ariba Procure-to-Pay function modules must be assigned to users who are authorized to run the transaction data export report transaction.

You must create the authorization object with the following details:

- **Object name**: F_KKMIGRAT
- **Description**: FI-CA IS Migration Workbench
- **Authorization Class**: FI Authorization

Fields:

- **EMG_ACTVT** = 1
- **EMG_FIRMA** = *
- **EMG_GROUP** = FIL

Creating roles for the authorization object

Context

You create roles for the authorization objects in SAP ERP to authorize only selected users to execute certain transactions.

Procedure

1. To create roles for the authorization object, run the transaction code PFCG. Enter the following values:
   - **Role**: for example, ZTRANSACTION_DATA
2. Click **Single Role**.
   - The Change Roles page appears.
3. Click the **Authorizations** tab and click **Change Authorization Data**.
   - The Choose Template page appears.
4. Click **Do not select templates**.
5. In the Manual selection of authorizations page, enter a name in the Authorization object text box. Ensure that you enter /ARB.

6. Click Continue.

The Change Role: Authorizations page appears.

7. Click the parent node, for example, ZTRANSACTION_DATA to expand it.

You can see the /ARB object class you created.

8. Click the second child Manually node.

The Activity and Program Name with Search Help child nodes appear.

9. Click the Program Name child node.

The Field Values dialog box appears.

10. In the Field values dialog box, ensure that you have the following entries:

   - Object: /ARBA/PROG
   - Field name: PROGRAM

11. In the Value Intrvl section, enter all the transaction data report names in the From column.

12. Save the entries and regenerate the profiles created.

13. In the Change Roles page, you must see the profiles assigned to the role.

   The Authorization tab turns to Green color to indicate that the roles have been created successfully.

14. In the Roles tab, copy the role, for example, ZTRANSACTION_DATA. Go to SU01, click the Roles tab and assign the role to the user.

15. Save the changes.

   Now, you can export the transaction data task.

Importing supplier data

<table>
<thead>
<tr>
<th>Function Module</th>
<th>/ARBA/VENDOR_EXPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSV Files</td>
<td>SupplierConsolidated.csv</td>
</tr>
<tr>
<td></td>
<td>PurchaseOrgSupplierCombo.csv</td>
</tr>
</tbody>
</table>

When you import the supplier data, Ariba Procurement Solution integrated with SAP downloads supplier data in the SupplierConsolidated.csv for the following:

- Suppliers available in the LFA1 (Vendor Master Table)
- Suppliers that are not marked for deletion for a given PORG
- Suppliers that do not have a centrally imposed purchasing block
- Suppliers that are not blocked for any function

The /ARBA/SYSTID_MAPS table stores values for the systemID and vendor. Before you import the supplier data, ensure that you maintain the correct values in the SystemID and Vendor fields in this table. When you import the supplier data, the values for the SystemID field is imported in the SupplierConsolidated.csv file.

Here is an example of how you can download a subset of the master data for suppliers by specifying the following parameters in the /ARBA/TVARV table.
• **Variable Name**: /ARBA/VENDOR_ONLY_EXPORT  
• **Field Name**: LIFNR  
• **Selection cat.**: S  
• **Number**: 0

In the Change View “TVARV generalized for fields”: Details section, you can enter the following values:

- **Char20**: VSAP  
- **Char20**: PSAP  
- **INCL/EXCL**: I  
- **Option**: EQ  
- **Selection value**: 0000001001  
- **Selection value**: Blank

**PurchaseOrg Supplier Combo**

Ariba Procurement Solution integrated with SAP downloads supplier data in the <PurchaseOrgSupplierCombo.csv> for the following:

- Suppliers available in the LFM1 (Vendor Master Record Purchasing Organization Data)  
- Suppliers that are not marked for deletion for a given PORG  
- Suppliers that do not have a centrally imposed purchasing block

**Supplier Location Data**

<table>
<thead>
<tr>
<th>Function Module</th>
<th>/ARBA/SUPPLIERLOCATION_EXPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSV File</td>
<td>SupplierLocationConsolidated.csv</td>
</tr>
</tbody>
</table>

The supplier location data contains two parameters that you maintain in the /ARBA/TVARV table: /ARBA/SL_VENDOR_ADDRESS and /ARBA/SL_PARTNER_TYPE.

- You must maintain at least one of these parameters in the /ARBA/TVARV table while downloading information for supplier locations. If you do not maintain one of these parameters, an error occurs.  
- **PARAMETER NAME**: /ARBA/SL_PARTNER_TYPE

To maintain the partner type filter for the supplier location:

- **Variable Name**: /ARBA/SL_PARTNER_TYPE  
- **Field Name**: PARVW  
- **Selection cat.**: S  
- **Number**: 0

In the Change View “TVARV generalized for fields”: Details section, here is an example of how you can maintain the following values:

- **Char20**: VSAP  
- **Char20**: PSAP
• **INCL/EXCL:** I
• **Option:** EQ
• **Selection value:** For example, OA. When you specify 'OA', Ariba Procurement Solution integrated with SAP downloads all the supplier locations for the supplier. You can maintain the partner type based on the configuration you have maintained for the partner function for a vendor.
• **Selection value:** Blank

**PARAMETER NAME: /ARBA/SL_VENDOR_ADDRESS**

To maintain the addresses filter for the supplier location:

• **Variable Name:** /ARBA/SL_VENDOR_ADDRESS
• **Field Name:** Blank
• **Selection cat.:** P
• **Number:** 0

Here is an example of how you can maintain values in the **TVARV generalized for fields** section:

• **Char20:** VSAP
• **Char20:** PSAP
• **INCL/EXCL:** I
• **Option:** EQ
• **Selection value:** ‘X’
  When you specify ‘X’, Ariba Procurement Solution integrated with SAP considers the supplier as the supplier location and downloads all addresses for the supplier location. You can maintain the partner type based on the configuration you have maintained for the partner function for a vendor.
• **Selection value:** Blank

You can maintain both the parameters in the **/ARBA/TVARV table.**

Here is how Ariba Procurement Solution integrated with SAP processes the supplier location. For example, the following table displays the values associated to the ABC Supplier:

<table>
<thead>
<tr>
<th>Supplier</th>
<th>Purchase Organization</th>
<th>Supplier Location 1</th>
<th>Supplier Location 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC Supplier</td>
<td>1000</td>
<td>1004</td>
<td>1005</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>1006</td>
<td>1007</td>
</tr>
</tbody>
</table>

**Case 1:** When you import the supplier location data, Ariba Procurement Solution integrated with SAP displays the Supplier Location for ABC supplier as 1004, 1005, 1006 and 1007 in the downloaded CSV file.

**Case 2:** When ABC Supplier is blocked for Purchase Organization 1000, Ariba Procurement Solution integrated with SAP only displays the Supplier Location as 1006 and 1007 in the downloaded CSV file.

**Note**

• Ariba Procurement Solution integrated with SAP does not support a supplier location linked to multiple vendors.
• When you do not maintain filters for **/ARBA/SL_VENDOR_ADDRESS** and **/ARBA/SL_PARTNER_TYPE**, Ariba Procurement Solution integrated with SAP displays an error stating that you have not maintained filters for the Supplier Location.
Remittance Location Data

<table>
<thead>
<tr>
<th>Function Module</th>
<th>/ARBA/REMITTANCE_LOCATION_EXPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSV File</td>
<td>RemittanceLocationConsolidatedExport.csv</td>
</tr>
</tbody>
</table>

You can maintain filters while downloading remittance locations by specifying the following parameters in the /ARBA/TVARV table.

- You can maintain both the /ARBA/RL_VENDOR_ADDRESS and /ARBA/RL_PARTNER_TYPE parameters. However, you must maintain at least one of the parameters while downloading the remittance information.
- You can download supplier as remittance location.
- When you want to download the supplier as the remittance location, you must maintain the /ARBA/RL_VENDOR_ADDRESS filter.
- **PARAMETER NAME: /ARBA/RL_PARTNER_TYPE**

To maintain the remittance partner type filter, ensure that you maintain the following values:

- **Variable Name:** /ARBA/RL_PARTNER_TYPE
- **Field Name:** PARVW
- **Selection cat.:** S
- **Number:** 0

Here is an example of how you can maintain values in the TVARV generalized for fields section:

- **Char20:** VSAP
- **Char20:** PSAP
- **INCL/EXCL:** I
- **Option:** EQ
- **Selection value:** IP

When you specify ‘IP’ as the selection value, Ariba Procurement Solution integrated with SAP considers the Remittance Location (LIFN2) with the Supplier Location (LIFNR) from the WYT3 table and downloads all the address information for the remittance location.

You can maintain the partner type based on the configuration you have maintained for the partner function for a vendor.

- **PARAMETER NAME: /ARBA/RL_VENDOR_ADDRESS**

To maintain the remittance address filter, ensure that you maintain the following values:

- **Variable Name:** /ARBA/RL_VENDOR_ADDRESS
- **Field Name:** Blank
- **Selection cat.:** P
- **Number:** 0

Here is an example of how you can maintain values in the TVARV generalized for fields section:

- **Char20:** VSAP
- **Char20:** PSAP
- **INCL/EXCL:** I
- **Option:** EQ
- **Selection value:** ‘X’

When you specify ‘X’ as the selection value, Ariba Procurement Solution integrated with SAP considers the supplier as the remittance location and downloads all the address details for the supplier.
You can maintain the partner type based on the configuration you have maintained for the partner function for a vendor.

You can have the following combinations for the supplier location and remittances location:

<table>
<thead>
<tr>
<th>Supplier Location Address</th>
<th>Supplier Location Partner Type</th>
<th>Remittance Location Address</th>
<th>Remittance Location Partner Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/SL_VENDOR_ADDRESS</td>
<td>-</td>
<td>Maintain</td>
<td>-</td>
</tr>
<tr>
<td>Maintain</td>
<td>-</td>
<td>Maintain</td>
<td>Maintain</td>
</tr>
<tr>
<td>Maintain</td>
<td>Maintain</td>
<td>Maintain</td>
<td>Maintain</td>
</tr>
<tr>
<td>Maintain</td>
<td>Maintain</td>
<td>Maintain</td>
<td>Maintain</td>
</tr>
<tr>
<td>Maintain</td>
<td>Maintain</td>
<td>Maintain</td>
<td>Maintain</td>
</tr>
</tbody>
</table>

Note

- When you maintain only the filter for the supplier location address /ARBA/SL_VENDOR_ADDRESS and do not maintain the supplier location and remittance location for the supplier, Ariba Procurement Solution integrated with SAP considers the supplier as the supplier location and remittance location.
- When you maintain the supplier location but do not maintain the remittance location, Ariba Procurement Solution integrated with SAP considers the supplier location as the remittance location.
- When you do not maintain filters for /ARBA/RL_VENDOR_ADDRESS and /ARBA/RL_PARTNER_TYPE, Ariba Procurement Solution integrated with SAP displays an error stating that you have not maintained filters for the Remittance Location.

PARAMETER NAME: /ARBA/RL_BANK_DETAILS

To maintain the bank details filter, ensure that you maintain the following values:

- Variable Name: /ARBA/RL_BANK_DETAILS
- Field Name: Blank
- Selection cat.: P
- Number: 0

Here is an example of how you can maintain values in the TVARV generalized for fields section:

- Char20: VSAP
- Char20: PSAP
- INCL/EXCL: I
- Option: EQ
- Selection value: 'X

Ariba procurement solutions integrated with SAP processes remittance locations as follows:

<table>
<thead>
<tr>
<th>Maintain filter for /ARBA/RL_BANK_DETAIL</th>
<th>Downloads Bank Details for Remittance Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintain</td>
<td>Yes</td>
</tr>
<tr>
<td>Do not maintain</td>
<td>No</td>
</tr>
</tbody>
</table>
Here is an example of how the supplier location and remittance location is mapped for a vendor:

Vendor 1050 has 1001 and 1002 as supplier locations. The remittance location associated to Vendor 1050 is 2000. This remittance location is associated to two payment banks - Dresden (10050033) and Hypo (19652993).

When you download the SupplierLocationConsolidated.csv, you can see 1001 and 1002 as the LocationID for VendorID 1050. The RemittanceLocationConsolidated.csv displays the two payment banks for both the 1001 and 1002 supplier locations.

**Note**

SAP ERP allows you to specify multiple bank accounts for the remittance location for a supplier. However, Ariba Procurement Solution does not support multiple bank accounts having the same payment method for a remittance location.

**Limitations for supplier location and remittance location**

When a supplier is either marked for Block or Delete and you run the master data for suppliers, Ariba Procurement Solution integrated for SAP does not extract the supplier, supplier purchase org, supplier location, or remittance location.

However, if the supplier is not marked for Block or Delete and active, but the supplier location or remittance location is either marked for Block or Delete, Ariba Procurement Solution integrated for SAP processes the information as follows:

- When you run the full load master data, Ariba Procurement Solution integrated for SAP downloads the:
  - Supplier information
  - Supplier purchase org, supplier location, or remittance location in the SupplierLocationConsolidated.csv and RemittanceLocationConsolidated.csv files. This information is downloaded as the supplier/vendor is active and not marked for Block or Delete.

- When you run the incremental master data:
  - Downloads information on the supplier, supplier location, or remittance location that has been changed recently for the supplier. The changed information is only available in the first run of the incremental load in the SupplierLocationConsolidated__Delete.csv files and RemittanceLocationConsolidated__Delete.csv files.

For example, you have Supplier 1000, having Supplier Location 2000 and Remittance Location 3000.

**Use Case 1**

1. Supplier 1000 is marked for Block or Delete.
2. When running the full load master data, Supplier 1000, Supplier Location 2000, and Remittance Location 3000 are not available in the SupplierLocationConsolidated.csv and RemittanceLocationConsolidated.csv files.
3. When you run the incremental load for suppliers, Supplier 1000 is available as blocked or deleted in the SupplierConsolidated_Delete.csv file.

**Use Case 2**

1. Supplier 1000 is not marked for Block or Delete.
2. Supplier Location 2000 or Remittance Location 3000 are marked for Block or Delete.
3. When running the full load master data, Supplier 1000 is extracted. However, Supplier Location 2000, and Remittance Location 3000 are also extracted and available in the `SupplierLocationConsolidated.csv` and `RemittanceLocationConsolidated.csv` files.

4. When running the incremental load for suppliers, Supplier Location 2000, and Remittance Location 3000 are extracted and available in the `SupplierConsolidated_Delete.csv` file.

**Note**

Although SAP ERP supports the Mark for Block or Delete functionality, Ariba Procurement Solution integrated for SAP does not support the Mark for Block or Delete functionality. When you run the master data for suppliers, this information about the supplier location or remittance location that has been blocked or deleted is still available in the .csv files.

---

**Importing payment terms data**

Payment terms indicate the negotiated discount between a buying organization and supplier for a specified number of days before payment is due.

<table>
<thead>
<tr>
<th>Function Module</th>
<th>/ARBA/PAYMENTTERM_EXPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSV File</td>
<td>PaymentTermsConsolidated.csv</td>
</tr>
</tbody>
</table>

**Note**

- When you import the payment terms data, Ariba Procurement Solution integrated with SAP imports only the Normal Payment Terms; the Installment Payment terms are excluded.
- `/ARBA/PAYMENT_TERMS` is the parameter to specify filters for payment terms.

For information on ERP considerations for importing payment terms and adding payment terms using a CSV file, see the *Ariba Invoicing and Payment Data Import and Administration Guide*.

---

**Importing user data and user group data**

<table>
<thead>
<tr>
<th>Function Modules</th>
<th>/ARBA/USER_EXPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>/ARBA/USER_GROUP_EXPORT</td>
</tr>
<tr>
<td>CSV Files Downloaded</td>
<td>UserConsolidated.csv</td>
</tr>
<tr>
<td></td>
<td>GroupConsolidated.csv</td>
</tr>
</tbody>
</table>

- The table `/ARBA/USR_GRP` must contain all the groups (SAP ERP roles) for users. If the user group is not maintained in this table, the user and the related user groups are not exported to the Ariba procurement solution system.
- Create user groups in the Ariba Procurement Solution with the SAP ERP role names using the import event *Import User Group Mapping*. 
• Import the users in the Ariba Procurement Solution using the import event Import User to Group Mapping Data.

You can specify the following parameters in the program /ARBA/MASTER_DATA_EXPORT to specify filters for user and user group data export task:

  ● /ARBA/USER_EXPORT
  ● /ARBA/USER_GROUP_EXPORT

**Note**

If you add fields to the customized RFC structure, add this field and the corresponding column name to the table /ARBA/FIELD_MAP to have a preferred column name in the output csv files. If you do not add this field, Ariba Procurement Solution integrated with SAP ERP displays the SAP ERP field name for that column available in the Ariba Procurement Solution. For more information, see Simplified master data integration [page 89].

When you export user data, only the information available in the transaction code SU01 is exported to the Ariba Procurement Solution. The HR information for the cost center, company code and the locale is also exported. The locale is the concatenation of HR nationality plus the HR Language (info type 0002). Ariba Procurement Solution integrated with SAP ERP does not import information about users maintained in the HR Info tables.

### Support integration for outline agreements in SAP ERP

Buyers can create contracts from outline agreements on their SAP ERP system and send them to the Ariba procurement solutions integrated with SAP. ERP buyers can export contract information to the Ariba procurement solutions using CSV files in the Ariba Integration Toolkit or directly without the Ariba Integration Toolkit for the following:

  ● Single line contracts
  ● Multiple line contracts
  ● Service line contracts
  ● Contracts containing split-accounting
  ● Contracts containing tiered pricing
  ● Contracts containing term pricing

### Requirements for outline agreements in SAP ERP

Before you import contract information, you must ensure that the master data information for the user, company code, plant, UOM, and accounting information on your ERP ECC system already exists on the Ariba procurement solution system.

You must maintain the parameter /ARBA/TEMP_DIRECTORY in the /ARBA/TVARV table before you import the contract information. For more information, see [page 69].

You must download the transport requests into your SAP ECC system and maintain the required parameters. For more information on the parameters, see Additional Information [page 99].
Limitations for outline agreements in SAP ERP

- The Ariba procurement solution integrated with SAP does not support the “Update” operation.
- Contracts containing both parent and child line items are not exported with this hierarchy. All child line items in contracts are exported as individual line items.
- You cannot export contracts for all the term types; only item-level containing specific items from supplier is supported.
- No support is available for split by amount accounting information. It also does not support the multi-line contract with duplicate material numbers.
- The following are the limitations for the term and tiered pricing:
  - The Ariba procurement solution integrated with SAP supports only two condition types: “PRICE_CONDITION_TYPE”, and “DISCOUNT_CONDITION_TYPE” However, the same line item cannot contain both these condition types.
  - The Term-based pricing supports only Discount Percentage and Gross Price. The Gross Price condition type is considered as the unit price. Only the Scale Type “From” is supported.
  - When a contract contains both tiered and term-based pricing, the Ariba procurement solution integrated with SAP only supports tiered-based pricing.
  - When the contract contains both the discount type and pricing type in a tiered pricing, only the pricing type values is used.
  - Only the current time validity period option in the tiered or term pricing is supported.

Additional references

- If you choose to use the Ariba Integration Toolkit to send the contract information, you require the following:
  Configure the `/ARBA/CONTRACT_EXPORT` report with the following parameters:
  - Enable the usage of the Ariba Integration Toolkit to export files to the procurement systems.
  - Parent Directory: Specify the absolute directory path to which the Ariba Data Transfer Tool exports the ZIP file.
  - Encoding: Specify the character encoding to zip and unzip the files.

  **Note**
  If you have not enabled the Ariba Integration Toolkit, the ERP system sends the contract files to the procurement systems.

- If you use the Ariba Integration Toolkit, the ERP buyer extracts the contract information in the following CSV files:
  - ContractHeader.csv
  - ContractItemInformation.csv
  - Accounting.csv
  - TermPricing.csv
  - TieredPricing.csv
- If the Ariba Integration Toolkit parameter is not enabled:
The function module /ARBA/DIRECT_CONNECT extracts the contract information as batch files and sends it to the procurement system as MIME type documents in to a temporary location.

Ensure that you maintain the following parameters in the configuration tables using the transaction code SM30:

- **ARBA/P2P_VENDOR**: Stores the vendors with contracts.
- **ARBA/LAST_CONTR**: This parameter is used for incremental exports. You must set the CONTRACTID and its associated CONTRACTTYPE. The parameter exports contracts with the CONTRACTID you set for the contract type. The CONTRACTID is a mandatory parameter; the CONTRACTTYPE is not mandatory.

If you maintain a different number range for different contract types, you must maintain the CONTRACTTYPE parameter here.

If you filter contracts by CONTRACTTYPE, then it is recommended that you use the CONTRACTTYPE parameter in this table.

- **ARBA/TVARV**: Use this parameter to filter by the document type adding the following entries in the /ARBA/TVARV table:
  - **Variable Name**: ARBA/CONTRACT_EXPORT
  - **Field Name**: BSART
  - **Selection cat.**: S
  - **Number**: 1

**TVARV generalized for fields**

- **Char20**: Blank
- **Char20**: Blank
- **INCL/EXCL**: I
- **Option**: EQ
- **Selection value**: WK
- **Selection value**: Blank
- **Variable Name**: ARBA/CONTRACT_EXPORT
- **Field Name**: BSART
- **Selection cat.**: S
- **Number**: 2

**TVARV generalized for fields**

- **Char20**: Blank
- **Char20**: Blank
- **INCL/EXCL**: I
- **Option**: EQ
- **Selection value**: MK
- **Selection value**: Blank

The /ARBA/TVARV table contains two additional parameters required only for term and tiered-based pricing:

- **PRICE_CONDITION_TYPE**
- **DISCOUNT_CONDITION_TYPE**
Note
The Ariba procurement solutions integrated with SAP supports only the price and discount type condition types.

Ensure that you maintain the following parameters for term based and tired pricing contracts:

- **Variable Name**: `PRICE_CONDITION_TYPE`
- **Field Name**: Blank
- **Selection cat.**: P
- **Number**: Blank

**TVARV generalized for fields**

- **Char20**: Blank
- **Char20**: Blank
- **INCL/EXCL**: I
- **Option**: EQ
- **Selection value**: PB00
- **Selection value**: Blank
- **Variable Name**: `DISCOUNT_CONDITION_TYPE`
- **Field Name**: Blank
- **Selection cat.**: P
- **Number**: Blank

**TVARV generalized for fields**

- **Char20**: Blank
- **Char20**: Blank
- **INCL/EXCL**: I
- **Option**: EQ
- **Selection value**: RA00
- **Selection value**: Blank

The report `/ARBA/CONTRACT_EXPORT` is available to export contracts to the procurement system using the Ariba Integration Toolkit or directly without the Ariba Integration Toolkit.

If you choose to export contracts directly without the Ariba Integration Toolkit, ensure that you maintain the following parameters in the table `/ARBA/HTTP_POST`:

- Realm
- SharedSecret
- URL

When you enable the Ariba Integration Toolkit, the following parameters are available:

- Parent Directory
- Encoding to be used
- Variant
- Partition
Importing master data to Federated Process Control-enabled systems

Master data integration between SAP ERP and the Ariba Procurement Solution supports export of master data from SAP ERP to the Ariba Procurement Solution in Federated Process Control-enabled environments. Buyers using the Ariba Procurement Solution integrated with SAP ERP can enable master data integration for Federated Process Control-enabled systems and specify the site, or realm, to which they want to export the master data.

You can implement this feature over Direct Connectivity or Mediated Connectivity integration methods.

To export master data to Federated Process Control-enabled systems, you must set the /ARBA/FPC_ENABLED parameter in the table /ARBA/TVARV to X. When you access the master data export program after you set the /ARBA/FPC_ENABLED to X, SAP ERP provides additional options in the window Export All Master Data Required for On-Demand Applications to help you specify settings for exporting master data to Federated Process Control-enabled systems.

You can specify the name of the site to which you want to export the master data in the Realm field of the Export All Master Data Required for On-Demand Applications window. In a Federated Process Control-enabled environment, you can export the same set of master data to multiple sites.

You can export the following master data from SAP ERP to Federated Process Control-enabled systems:

- Organizational Data
- Accounting Data
- Cross/Application Configuration Data
- Supplier Related Data
- User Related Data

SAP ERP supports both full load and incremental load export of the master data.

Incremental export of Master Data to Federated Process Control-enabled systems

SAP ERP uses a combination of the following three variables to ensure that the incremental master data is available to all sites to which you want to do incremental exports:

- Site information
- Date and time stamp
- Retention Period

When you do an incremental export to a Federated Process Control-enabled system, SAP ERP tags each of the records that you export with the date and time of the export and the name of the site to which you export the record. SAP ERP uses the site information and date and time stamp to identify the records to be exported when a new incremental export is triggered.

You can configure the /ARBA/CHANGELOG_RETENTION_DAYS in the table /ARBA/TVARV to specify the number of days you want SAP ERP to retain the change records in the table/ARBA/CHANGE_DOC. When the configured retention period expires, SAP ERP clears the change log.
**Note**

If you set the `/ARBA/FPC_ENABLED` parameter in the table `/ARBA/TVARV` to X and do not maintain a value for `/ARBA/CHANGECLOG_RETENTION_DAYS`, SAP ERP returns an error when you access the master data export program.

---

**Segregation of supplier and user master data elements for specific sites**

When you export user and supplier master data to a Federated Process Control-enabled system, you can segregate various elements of the user and supplier master data and send subsets or supersets based on the specific requirements. When you export user master data to a parent site, as configured in the table `/ARBA/AUTH_PARAM`, SAP ERP sends only those master data elements that are relevant for the parent site. If the realm is not marked as a parent realm in the table `/ARBA/AUTH_PARAM`, SAP ERP sends only those master data fields that are relevant for a child site.

For example, the user master data sent to a parent site does not contain accounting information, whereas the user master data sent to a child site contains accounting information.

When you choose user master data for export in non-Federated Process Control-enabled environment, SAP ERP sends both user and user group master data. However, in Federated Process Control-enabled environments, you can choose to export either the user or the user group master data, or both.

Similarly, you can choose to segregate the supplier master data into various master data elements based on the specific requirements of the site to which you export the supplier master data. The supplier master data segregation options include supplier location, supplier purchase organization, and remittance location.

The parent and child differentiation applies only to user and supplier master data events. The following table lists the .CSV files generated for the user and supplier master data events and their availability to parent and child sites:

<table>
<thead>
<tr>
<th>.CSV Filename</th>
<th>Applicable Site</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserConsolidated.csv</td>
<td>Parent and child sites.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong></td>
</tr>
<tr>
<td></td>
<td>The <code>UserConsolidated.csv</code> file for the parent realm contains only a subset of master data elements. For a list of master data elements that are exported to a parent site [User master data elements exported to a parent site][page 105].</td>
</tr>
<tr>
<td>SupplierConsolidated.csv</td>
<td>Parent and child sites</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong></td>
</tr>
<tr>
<td></td>
<td>The <code>SupplierConsolidated.csv</code> file for Federated Process Control-enabled sites contains a new field, <code>ImportCtrl</code>.</td>
</tr>
</tbody>
</table>
Prerequisites for importing master data to Federated Process Control-enabled systems

- SAP ERP 6 EhP 0 SP14 or later versions.
- Install the latest transports. If you want to use Mediated Connectivity, import the latest TPZ files.
- Configure Direct or Mediated Connectivity settings.
- Run the database utility to update the enhancements made to the following tables:
  - `/ARBA/AUTH_PARAM`
  - `/ARBA/INCR_DTTIM`
  - `/ARBA/TVARV`
  - `/ARBA/USR_GRP`
- Maintain values for the REALM, LOGICAL_PORT, and IS_PARENT fields in the `/ARBA/AUTH_PARAM` table. The logical port field contains the logical port that you configure as part of the SOAMANAGER configuration.

  **Note**

  If you are using Mediated Connectivity, ensure that the logical port field is empty.

- Maintain supplier and remittance location in the `/ARBA/TVARV` table.
- Maintain the `/ARBA/USR_GRP` table.
- If you are upgrading from a non-Federated Process Control system to Federated Process Control-based system, maintain the last run time for the user master data export in the `/ARBA/INCR_DTTIM` table.
- If you are using Mediated Connectivity, in SAP Process Integration create a business system each for the parent and child sites you have configured.

Limitations for importing master data to Federated Process Control-enabled systems

- Integrated Toolkit-based connections are not supported for master data integration between SAP ERP and the Ariba Procurement Solution in Federated Process Control-enabled environments.
- The remittance location data is linked to the supplier location. When you export remittance location, even if you choose only the remittance location for export, SAP ERP extracts the corresponding supplier location information too. When you export supplier location in the incremental load mode after exporting remittance...
location, the export does not contain those supplier location records that SAP ERP extracted while exporting remittance location master data.

- Remittance location option is not available for Ariba Procure-to-Order users.

**User master data elements exported to a parent site**

The `UserConsolidated.csv` file exported to a parent site contains the following user master data elements:

- `UniqueName`
- `Name`
- `EmailAddress`
- `DefaultCurrency.UniqueName`
- `LocaleID.UniqueName`
- `TimeZoneID`
- `Phone`
- `Fax`
- `Supervisor.UniqueName`
- `Supervisor.PasswordAdapter`
- `Title Prefix`
- `FirstName`
- `LastName`
- `MobileNumber`
- `Address1`
- `Address2`
- `City`
- `Country`
- `State`
- `PostalCode`
- `EmployeeID`
- `DeactivationDate`
- `RoleName`

**Maintaining table entries**

Before you can export master data from SAP ERP to the Ariba Procurement Solution in Federated Process Control-enabled environments, you must maintain the table parameters listed in the following sections:

- `/ARBA/AUTH_PARAM` [page 106]
- `/ARBA/TVARV` [page 106]
- `/ARBA/USR_GRP` [page 109]
- `/ARBA/INCR_DTTIM` [page 110]

You can use the transaction code `sm30` to access tables. To maintain a table, enter the name of the table in the `Table/View` field of the Maintain Table Views: Initial Screen, and click Maintain.
In the table `/ARBA/AUTH_PARAM`, maintain the following parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution</td>
<td>Name of the Ariba solution that you integrate with SAP ERP.</td>
<td>Ariba</td>
</tr>
<tr>
<td>Ariba Realm ID</td>
<td>The name of your company as configured in the Ariba application.</td>
<td>s4All</td>
</tr>
<tr>
<td>Wait Time</td>
<td>Number of minutes to wait before SAP ERP attempts to reconnect with the Ariba application.</td>
<td>02</td>
</tr>
<tr>
<td>Logical Port</td>
<td>Name of the logical port that you configure as part of the SOAMANAGER configuration.</td>
<td>ARIBA_PARENT</td>
</tr>
<tr>
<td>Is Parent</td>
<td>Whether the realm is a parent realm.</td>
<td>Checked or unchecked</td>
</tr>
<tr>
<td></td>
<td>• Checked Is Parent indicates that the corresponding realm is a parent realm.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Unchecked Is Parent indicates that the corresponding realm is a child realm.</td>
<td></td>
</tr>
</tbody>
</table>

Maintain the following parameters in the table `/ARBA/TVARV`:

- `/ARBA/EXTERNAL_SID` is a mandatory parameter that is required for the Direct Connectivity configuration.

<table>
<thead>
<tr>
<th>Field</th>
<th>Values for</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable Name</td>
<td>/ARBA/EXTERNAL_SID</td>
</tr>
<tr>
<td>Field Name</td>
<td>--</td>
</tr>
<tr>
<td>Selection Category</td>
<td>P</td>
</tr>
<tr>
<td>Number</td>
<td>0</td>
</tr>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
</tbody>
</table>
**/ARBA/FPC_ENABLED** is a mandatory parameter for exporting master data to Federated Process Control-enabled systems. The value you set for this parameter indicates whether master data export is enabled for Federated Process Control-enabled systems or not.

<table>
<thead>
<tr>
<th>Field</th>
<th>Values for FPC_ENABLED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable Name</td>
<td>/ARBA/FPC_ENABLED</td>
</tr>
<tr>
<td>Field Name</td>
<td>--</td>
</tr>
<tr>
<td>Selection Category</td>
<td>P</td>
</tr>
<tr>
<td>Number</td>
<td>0</td>
</tr>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
<tr>
<td>INCL/EXCL</td>
<td>I</td>
</tr>
<tr>
<td>Options</td>
<td>EQ</td>
</tr>
<tr>
<td>Selection Value</td>
<td>X</td>
</tr>
<tr>
<td>Selection Value</td>
<td>--</td>
</tr>
</tbody>
</table>

**/ARBA/CHANGELOG_RETENTION_DAYS** is a mandatory parameter for exporting master data to Federated Process Control-enabled systems. The value you set for this parameter indicates the number of days that you want SAP ERP to retain change log for incremental export.

<table>
<thead>
<tr>
<th>Field</th>
<th>Values for CHANGELOG_RETENTION_DAYS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable Name</td>
<td>/ARBA/CHANGELOG_RETENTION_DAYS</td>
</tr>
<tr>
<td>Field Name</td>
<td>--</td>
</tr>
<tr>
<td>Selection Category</td>
<td>P</td>
</tr>
<tr>
<td>Number</td>
<td>0</td>
</tr>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
</tbody>
</table>
### Values for CHANGELOG RETENTION DAYS

<table>
<thead>
<tr>
<th>Field</th>
<th>Values for CHANGELOG RETENTION DAYS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
<tr>
<td>INCL/EXCL</td>
<td>I</td>
</tr>
<tr>
<td>Options</td>
<td>EQ</td>
</tr>
<tr>
<td>Selection Value</td>
<td>2</td>
</tr>
</tbody>
</table>

**Note**

### Values for TEMP_DIRECTORY

<table>
<thead>
<tr>
<th>Field</th>
<th>Values for TEMP_DIRECTORY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable Name</td>
<td>/ARBA/TEMP_DIRECTORY</td>
</tr>
<tr>
<td>Field Name</td>
<td>--</td>
</tr>
<tr>
<td>Selection Category</td>
<td>P</td>
</tr>
<tr>
<td>Number</td>
<td>0</td>
</tr>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
<tr>
<td>INCL/EXCL</td>
<td>I</td>
</tr>
<tr>
<td>Options</td>
<td>EQ</td>
</tr>
<tr>
<td>Selection Value</td>
<td>Number of days you want to retain the change log. The location to which you want to extract the master data. For example, . \MASTERDATA.</td>
</tr>
</tbody>
</table>

### Values for RL_PARTNER_TYPE

<table>
<thead>
<tr>
<th>Field</th>
<th>Values for RL_PARTNER_TYPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable Name</td>
<td>/ARBA/RL_PARTNER_TYPE</td>
</tr>
<tr>
<td>Field Name</td>
<td>PARVW</td>
</tr>
<tr>
<td>Field</td>
<td>Values for RL_PARTNER_TYPE</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>Selection Category</td>
<td>S</td>
</tr>
<tr>
<td>Number</td>
<td>0</td>
</tr>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
<tr>
<td>INCL/EXCL</td>
<td>I</td>
</tr>
<tr>
<td>Options</td>
<td>EQ</td>
</tr>
<tr>
<td>Selection Value</td>
<td>IP</td>
</tr>
<tr>
<td>Selection Value</td>
<td>--</td>
</tr>
</tbody>
</table>

/ARBA/SL_PARTNER_TYPE is a mandatory parameter for supplier master data export. The value you set for this parameter indicates the supplier location.

<table>
<thead>
<tr>
<th>Field</th>
<th>Values for SL_PARTNER_TYPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable Name</td>
<td>/ARBA/SL_PARTNER_TYPE</td>
</tr>
<tr>
<td>Field Name</td>
<td>PARVW</td>
</tr>
<tr>
<td>Selection Category</td>
<td>S</td>
</tr>
<tr>
<td>Number</td>
<td>0</td>
</tr>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
<tr>
<td>Char 20</td>
<td>--</td>
</tr>
<tr>
<td>INCL/EXCL</td>
<td>I</td>
</tr>
<tr>
<td>Options</td>
<td>EQ</td>
</tr>
<tr>
<td>Selection Value</td>
<td>OA</td>
</tr>
<tr>
<td>Selection Value</td>
<td>--</td>
</tr>
</tbody>
</table>

/ARBA/USR_GRP

In the table /ARBA/USR_GRP, maintain the following parameters:
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution</td>
<td>Name of the Ariba solution that you integrate with SAP ERP.</td>
<td>Ariba</td>
</tr>
<tr>
<td>Role</td>
<td>User role.</td>
<td></td>
</tr>
<tr>
<td>Realm</td>
<td>The name of your company as configured in the Ariba application.</td>
<td>S4all</td>
</tr>
</tbody>
</table>

/ARBA/INCR_DTTIM

For user master data integration, maintain the following values in the table /ARBA/INCR_DTTIM:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Export Name</td>
<td>Name of the export.</td>
<td>USER</td>
</tr>
<tr>
<td>Realm</td>
<td>Name of the Ariba procurement realm to which the data was exported.</td>
<td>P2P</td>
</tr>
<tr>
<td>Timestamp</td>
<td>Date and time of the export. Configure an initial value of 0. After every successful export, the timestamp value gets updated based on the timestamp the Ariba Procurement Solution sends to SAP ERP. If you are moving from a non-FPC implementation to an FPC-based implementation, specify the timestamp of the last successful run of the user master data export.</td>
<td>0</td>
</tr>
</tbody>
</table>

How to import master data in Federated Process Control-enabled environments

Prerequisites

- Maintain necessary table entries as explained in Maintaining table entries [page 105].
Context

In Federated Process Control-enabled environments, you can specify the site to which you want to import the master data from SAP ERP. When you import user and supplier master data in a Federated Process Control-enabled environment, you can segregate user and supplier master data into specific elements that you want to export to specific sites.

Note

Always export the master data to the parent realm before you export the master data to child realms.

Procedure

1. Run the transaction code /ARBA/MASTER_DATA.
   
   The Export All Master Data Required for On-Demand Application screen appears.

2. Choose one of the following options from the Specify the Scope of the Procurement Solutions section:
   ○ Procure-to-Pay
   ○ Procure-to-Order

3. From Connectivity Methods, choose Direct Connectivity.

   Note

   Integration Toolkit-based connections are not supported for master data integration with Federated Process Control-enabled systems.

4. From Load Options, choose one of the following options:
   ○ Full Load
   ○ Incremental Load

5. In the Realm text box, enter the name of the site to which you want to send the master data.

6. From the Select Master Data for Export list of check boxes, check the master data elements that you want to export.

   When master data integration for Federated Process Control-enabled systems is enabled, the following additional check boxes are available for the Supplier and User master data groups:

   For the Supplier master data:
   ○ Supplier Location
   ○ Remittance Location

   Note

   The additional check boxes appear after you check Supplier.

   The Remittance Location option is not available for Procure-to-Order users.

   ○ Supplier PurchaseOrg

   For the User master data:
Configuring Direct Connectivity

You can configure Direct Connectivity to enable seamless integration of master data between Ariba procurement solutions and SAP ERP. You can use the SOAMANAGER configuration to set up direct connectivity between Ariba procurement solutions and SAP ERP.

Prerequisites

- Install the latest transports as explained in How to import the Ariba Components.
- Maintain the table parameters as listed in Maintaining table entries.
- Install the SSL client certificates as explained in How to install the SSL client certificate for HTTPS connection.
- Create a logical file path for the ISMW_ROOT. For more information, see How to create a logical file path.
- Specify the system ID of SAP ERP on the Service Manager Site Manager Customer Site Master Data Manager External System Configuration Create External System window of the Ariba application.
- Based on the security model you choose, configure one of the following:
  - If you are using Shared Secret-based authentication, configure the Shared Secret settings in the Service Manager Site Manager Customer Site Integration Manager Integration Toolkit Security window of the Ariba application.
  - If you are using the certificate-based authentication, install the client certificates and configure the certificate settings at the Service Manager Site Manager Customer Site Integration Manager Integration Toolkit Security window of the Ariba application. For more information on installing the certificate, see How to configure the client certificate for certificate-based authentication.

How to import the Ariba Components

Context

To be able to export master data from SAP ERP to Ariba Procurement Solution in Federated Process Control-enabled systems, you must import the latest transports to SAP ERP and the design objects to SAP Process Integration.
Procedure

1. Go to connect.ariba.com and log in using your user ID and password. If you do not have a User ID and Password for Ariba Connect, contact your Ariba account executive.
2. On the Home tab, in the Product Summary > On-Demand page, click Ariba Cloud Integration.
3. In the Integration Tools section, click Integration tools for Ariba Procure-to-Pay.
4. Go to the Integration Tools for SAP section and then click ABAP Transports for SAP ERP. If the list does not contain this entry, contact your Ariba account executive.
5. Click Download and specify a location on your hard drive to download and save the ZIP file.
6. Extract the transport files from the Ariba_P2P_SAP_Transports_CI8_Aug2016.zip to the appropriate locations on your SAP ERP.
7. If you want to use Mediated Connectivity, go back to the Integration Tools for SAP section and click SAP NetWeaver PI Mapping. If the list does not contain this entry, contact your Ariba account executive.
8. Click Download and specify a location on your hard drive to download and save the ZIP file.
9. Extract and import the relevant TPZ files into the SAP Process Integration.

Incremental load events

You can run the Incremental change transactions either using the Ariba Integration Toolkit or directly while running the master data events to pick up the new, updated, or deleted changes.

You can run the incremental change transactions for the following master data tasks:

- User and User Group information
  For more information, see User and User Group Incremental Load [page 122].
- Supplier-related information that includes supplier location and supplier remittance location
  For more information, see Supplier Incremental Load [page 124].
- General Ledger
  For more information, see General Ledger Incremental Load [page 118].
- Internal Order and Company Code
  For more information, see Internal Order Incremental Load [page 121].
- Cost Center
  For more information, see Cost Center Incremental Load [page 116].
- WBS Element and Company Code
  For more information, see WBS Element Incremental Load [page 122].
- Assets
  For more information, see Asset Incremental Load [page 114].

The function module /ARBA/CHANGE_DOC stores details of all the changes to the master data in the /ARBA/CHANGE_DOC table.
The `/ARBA/INCR_DTTIM` table captures the time stamp for the last run master data export tasks.

The Ariba Procurement Solution integrated with SAP ERP supports the following incremental change transactions operations for the master data export tasks:

<table>
<thead>
<tr>
<th>Object</th>
<th>Create</th>
<th>Change</th>
<th>Block</th>
<th>Unblock</th>
<th>Lock</th>
<th>Unlock</th>
<th>Delete</th>
<th>Undelete</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supplier</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N/A</td>
<td>N/A</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>User</td>
<td>Y</td>
<td>Y</td>
<td>N/A</td>
<td>N/A</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Asset</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N/A</td>
<td>N/A</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Cost Center</td>
<td>Y</td>
<td>Y</td>
<td>N/A</td>
<td>N/A</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>General Ledger</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N/A</td>
<td>N/A</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>WBS</td>
<td>Y</td>
<td>Y</td>
<td>N/A</td>
<td>N/A</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Internal Order</td>
<td>Y</td>
<td>Y</td>
<td>N/A</td>
<td>N/A</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>

### Asset incremental load

The asset incremental change transaction supports the following:

- Creating a new asset - AS01
- Changing an existing asset - AS02
- Blocking an existing asset - AS05
- Deleting an existing asset - AS06

The `/ARBA/CHANGE_DOC` table records all the changes for these transactions. All the created and changed assets are stored in the Asset.csv file. The Asset_Delete.csv file stores all the deleted changes.

The following parameters have been added to the `/ARBA/ASSET_EXPORT` function module:

- `INCREMENTAL`: This parameter is required for the full and incremental load.
- `CHANGE_DOC`: This parameter stores the changes that are fetched from the `/ARBA/CHANGE_DOC` table. This can be used to delete the processed records from the change log table.

The following is the configuration that your ERP system maintains for the asset incremental change transaction:

In SWO1, the business object `/ARBA/1022` contains the following events:

- AssetCreated
- AssetChanged
- AssetDeleted

In the SWEC, the change document object ANLA is assigned to the business object `/ARBA/1022`.

The following are the options maintained for the create, change, delete, and block events:

<table>
<thead>
<tr>
<th>Business Obj. Type</th>
<th>Event</th>
<th>Select Option</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/ARBA/1022</code></td>
<td>ASSETCREATED</td>
<td>On Create</td>
</tr>
<tr>
<td><code>/ARBA/1022</code></td>
<td>ASSETDELETED</td>
<td>On Delete</td>
</tr>
</tbody>
</table>
### Business Obj. Type

| /ARBA/1022 | AssetDeleted | On Change |
| /ARBA/1022 | AssetChanged | On Change |

**Note**

This option is also for block events.

---

**Note**

The Ariba Procurement Solution integrated with SAP ERP tracks only the changes to the fields in the change and block event; all other changes are ignored. The field restrictions for the change and block events track only the changes to the fields; all other changes are ignored.

The following field restrictions are available for change events:

![Change Condition: Field restriction for ANLA](image1)

The following field restrictions are available for block events:

![Change Condition: Field restriction for ANLA](image2)
In SWETYPV, the following options are assigned to the receiver function for each event:

- **Object Category**: BOR Object Type
- **Object Type**: Specify the extension value for each of the change object.
- **Event**: Specify the event type. For example, CHANGEGL.
- **Receiver Type**: CHANGE

The Linkage Setting (Event Receiver) section contains the following settings:

- **Receiver Call**: Function Module
- **Receiver Function Module**: /ARBA/CHANGE_DOC
- **Check Function Module**: /ARBA/CHANGE_DOC
- **Receiver Type Function Module**: Blank
- **Destination of Receiver**: Blank
- **Event Delivery**: Using tRFC (Default)
- **Linkage Activated check box**: Checked
- **Behavior Upon Error Feedback pull-down list**: System defaults
- **Receiver Status pull-down list**: No errors

## Cost center incremental load

The cost center change transaction supports the following:

- Creating a new cost center - KS01
- Changing an existing cost center - KS02
- Locking an existing cost center - KS02

**Note**

You cannot run the incremental delete load for cost centers.

The /ARBA/CHANGE_DOC table records all the changes for these transactions. All the created and changed information for cost centers is stored in the CostCenter.csv file. The CostCenter_Delete.csv file stores all the deleted changes.

The following parameters have been added to the /ARBA/COST_CENTER_EXPORT function module:

- **INCREMENTAL**: This parameter is required for the full and incremental load.
- **CHANGE_DOC**: This parameter stores the changes that are fetched from the /ARBA/CHANGE_DOC table. This can be used to delete the processed records from the change log table.

Your SAP ERP maintains the following configuration for the cost center incremental change transaction: In SWO1, the business object, /ARBA/0012 contains the following events:

- CostCenterCreated
- CostCenterChanged
- CostCenterDeleted

In the SWEC, the change document object event is assigned to the business object /ARBA/0012.
The following are the options maintained for the create, change, and delete events:

<table>
<thead>
<tr>
<th>Object Type</th>
<th>Event</th>
<th>Select Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/USER</td>
<td>CLONED</td>
<td>On Change</td>
</tr>
<tr>
<td>/ARBA/0012</td>
<td>COSTCENTERCREATED</td>
<td>On Create</td>
</tr>
<tr>
<td>/ARBA/0012</td>
<td>COSTCENTERDELETED</td>
<td>On Change</td>
</tr>
<tr>
<td>/ARBA/0012</td>
<td>COSTCENTERCHANGED</td>
<td>On Change</td>
</tr>
</tbody>
</table>

**Note**

This entry is for user export.

**Note**

This entry is for block events.

The Ariba Procurement Solution integrated with SAP ERP tracks only the changes to the fields in the change and block event; all other changes are ignored. The field restrictions for the change and block events track only the changes to the fields; all other changes are ignored.

The following field restrictions are available for change events:
The following field restrictions are available for block events:

In SWETYPV, following are the options assigned in the receiver function module /ARBA/CHANGE_DOC:

- **Object Category**: BOR Object Type
- **Object Type**: Specify the extension value for each of the change object.
- **Event**: Specify the event type. For example, COSTCENTERCHANGED.
- **Receiver Type**: CHANGE
  The Linkage Setting (Event Receiver) section contains the following settings:
  - **Receiver Call**: Function Module
  - **Receiver Function Module**: /ARBA/CHANGE_DOC
  - **Check Function Module**: /ARBA/CHANGE_DOC
  - **Receiver Type Function Module**: Blank
  - **Destination of Receiver**: Blank
  - **Event Delivery**: Using tRFC (Default)
  - **Linkage Activated check box**: Checked
  - **Behavior Upon Error Feedback pull-down list**: System defaults
  - **Receiver Status pull-down list**: No errors

**General ledger incremental load**

The general ledger incremental change transaction supports the following:

- Creating a new general ledger - FS01
- Changing an existing general ledger - FS02
- Blocking an existing general ledger - FS02
- Deleting an existing general ledger - FS02

The block general ledger supports the following operations:

- Block in chart of accounts
○ Block for creation  
○ Block for posting  
○ Block for planning 

● Block in company code  
○ Block for posting 

The delete general ledger supports the following operations:

● Deletion flag chart of accounts  
  ○ Mark for deletion  

● Deletion flag in company code  
  ○ Mark for deletion 

The /ARBA/CHANGE_DOC table records all the changes for these transactions. All the created and changed events are stored in the GeneralLedger.csv file. The GeneralLedger_Delete.csv file stores all the deleted changes.

The following parameters have been added to the /ARBA/GENERAL_LEDGER_EXPORT function module:

● INCREMENTAL: This parameter is required for the full and incremental load.
● CHANGE_DOC: This parameter stores the changes that are fetched from the /ARBA/CHANGE_DOC table. This can be used to delete the processed records from the change log table.

Your SAP ERP maintains the following configuration for the incremental load for general ledgers:

In the SWO1, the business object /ARBA/3066 contains the following events:

● CreateGL  
● ChangeGL  
● DeleteGL 

In SWEC, the change document object event is assigned to the business object /ARBA/3006.

The following are the entries for the create, change, and delete events:

<table>
<thead>
<tr>
<th>Object Type</th>
<th>Event</th>
<th>Select Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/3066</td>
<td>CREATEGL</td>
<td>On Create</td>
</tr>
<tr>
<td>/ARBA/3066</td>
<td>DELETEGL</td>
<td>On Change</td>
</tr>
<tr>
<td>/ARBA/3066</td>
<td>CHANGEGL</td>
<td>On Change</td>
</tr>
</tbody>
</table>

**Note**

This entry is also for block events.

The Ariba Procurement Solution integrated with SAP ERP tracks only the changes to the fields in the change and block event; all other changes are ignored. The field restrictions for the change and block events track only the changes to the fields; all other changes are ignored.
The following field restrictions are available for change events:

![Change Condition: Field restriction for SACH](image1)

The following field restrictions are available for block events:

![Change Condition: Field restriction for SACH](image2)

In the SWETYPV, the following options are assigned in the receiver function module /ARBA/CHANGE_DOC for each event:

- **Object Category**: BOR Object Type
- **Object Type**: Specify the extension value for each of the change object.
- **Event**: Specify the event type. For example, CHANGEGL.
- **Receiver Type**: CHANGE
  In the Linkage Setting (Event Receiver) section, specify the following:
- **Receiver Call**: Function Module
- **Receiver Function Module**: /ARBA/CHANGE_DOCS
- **Check Function Module**: /ARBA/CHANGE_DOCS
- **Receiver Type Function Module**: Blank
- **Destination of Receiver**: Blank
- **Event Delivery**: Using tRFC (Default)
- **Linkage Activated check box**: Checked
- **Behavior Upon Error Feedback pull-down list**: System defaults.
- **Receiver Status pull-down list**: No errors

**Internal order incremental load**

The internal order incremental change transaction supports the following:

- Creating an internal order - KO01
- Changing an existing internal order - KO02
- Locking an existing internal order - KO02
- Deleting an internal order - KO02

Changes for the internal order is maintained and tracked using the last run time stamp in the /ARBA/INCR_DTTIM table. All the created and changed assets are stored in the InternalOrder.csv and CompanyCodeIOCombo.csv files. The InternalOrder_Delete.csv and CompanyCodeIOCombo_Delete.csv files store all the locked and deleted changes.

The following parameter has been added to the /ARBA/INTERNAL_ORDER_EXPORT function module:

**INCREMENTAL**: This parameter is required for the full and incremental load.

You must configure the date and time in your SAP ERP to run the incremental load for internal orders.

**How to configure the date and time in the incremental load for internal orders**

**Procedure**

In the /ARBA/INCR_DTTIM table, ensure that you maintain the following entries:

- **MANDT**: Specify the number of the client SAP ERP systems.
- **ExportName**: IO
- **DTTIME**: Specify the date and time.
WBS element incremental load

The WBS incremental change transaction supports the following:

- Creating a WBS element - CJ11
- Changing a WBS element - CJ12
- Deleting a WBS element - CJ12

The /ARBA/INCR_DTTIM table stores the date and time stamp in the DTTIME column. However, the Ariba Procurement Solution integrated with SAP ERP does not export the time stamp value available in this column. To export the date stamp correctly, Ariba recommends that you maintain a zero value for the timestamp in the DTTIME column for the WBS element. For example, 20140916000000.

**Note**

The timestamp changes in the WBS element are not considered while exporting the incremental load. The Ariba Procurement Solution integrated with SAP ERP exports only the date stamp.

How to configure the date and time in the incremental load for WBS element

**Procedure**

In the /ARBA/INCR_DTTIM table, ensure that you maintain the following entries:

- **MANDT**: Specify the number of the client SAP system.
- **ExportName**: WBS
- **DTTIME**: Specify the date and time.

User and user group incremental load

The user and user group incremental change transaction supports the following:

- Changes to users maintained in SU01
- Changes to users HR data in PA transactions

Supported operations for change of users in SU01:

- Lock users
- Delete users permanently
- Change validity dates
- Add or delete Ariba groups
- Change addresses
Supported operations for change of users HR data:

- Create info type PA0105 (communications) with the user type as user ID, lock of PA0105, validity changes of PA0105
- Changes to organizational assignment PA0001
- Changes to personal data PA0002

The following are not supported for Users in SU01:

- The incremental changes for the time zone in the Defaults tab in the transaction code SU01 are not supported as there are no changes logged in SAP ERP.

The incremental changes for the Parameter tab in SU01 is not supported as there are no changes logged in SAP ERP. The Ariba Procurement Solution integrated with SAP ERP extracts only the Purchase Org., Purchase Group, Currency from the Parameters tab. In the Parameter tab, changes done to these fields are not recorded and hence we do not support an incremental load.

The following are not supported for the HR-PA transactions:

- SAP ERP does not support the delete, change, lock and change validity dates. Hence the Ariba Procurement Solution integrated with SAP ERP does not support the delete, lock, or validity changes of 0001 (Organization) and 0002 (Personal) info types.
- The delete of infotype 0105 (Communications) does not trigger any changes.
- Changes to the Organizational, Communication, or Personal infotypes are recorded only if there is a user to the HR link (Infotype 0105) (valid or locked).
- No field restrictions are available in the Organizational, Communication, or Personal infotypes. The Ariba Procurement Solution integrated with SAP ERP records all field changes.
- If the infotype 0105 (Communications) is locked, deleted, or not valid, no HR details is pulled in either the incremental or full load export task.
- Any addition or deletion to user groups in the SU01 Role tab is only maintained in SAP ERP.
- When user information has been changed, deleted or locked, the Ariba Procurement Solution integrated with SAP ERP does not validate if the user is already available in the Ariba Procurement Solution.
- If a user is locked in SAP ERP and you export the user, the user is removed from the Ariba Procurement Solution. However, the user that is locked still remains in SAP ERP. If you make changes to the locked user in SAP ERP and then run the incremental load, the Ariba Procurement Solution errors out while exporting the locked user to the Ariba Procurement Solution. This is because the locked user does not exist in the Ariba Procurement Solution.

How to configure the date and time in the incremental load for users

Prerequisites

- You must configure the date and time for users. See Configuring the Date and Time in the Incremental Load for Users [page 123].
- See Exporting User Data and User Group Data [page 97] for more information on the other prerequisites required for the user and user group incremental change transaction.
Procedure

In the /ARBA/INCR_DTTIM table, ensure that you maintain the following entries:

- **MANDT**: Specify the number of the client SAP ERP systems.
- **ExportName**: USER
- **DTTIME**: Specify the date and time.

Supplier incremental load

The supplier incremental change transaction supports the following:

- **Creating suppliers** - XK01
  - Extending suppliers to certain purchase organizations
- **Changing suppliers** - XK02
  - Changes to the supplier address fields that have been configured
  - Unblock for quality reasons
  - Unblock selected purchase organizations
  - Unblock all purchase organizations
  - Undo the delete flags for all areas
  - Create supplier location
  - Create remittance location
  - Delete supplier location
  - Delete remittance location
- **Delete or block suppliers** - XK02 (XK05/XK06)
  - Block for quality reasons
  - Block selected purchase organizations
  - Block all purchase organizations
  - Delete flags for all areas

**Note**

To extract deleted suppliers, check only **All areas**. If you check **All areas, Selected company code**, and/or **Selected purchasing organization**, then the suppliers are extracted not as deleted suppliers but as changed suppliers.

The supplier incremental load uses the /ARBA/VENDOR_INCREMENT_EXPORT function module and supports the following events:

- VENDORCREATED
- VENDORCHANGED
- VENDORDELETED

Prerequisites

- The /ARBA/CHANGE_DOC table captures all the changes for the supplier incremental load event.
- The business object LFA1 is extended to the sub object /ARBA/LFA1. The change or delete events for supplier, supplier location, and remittance location is captured using the business object and event linkages.
● In the transaction code, SWETYVP, create a link between the function module /ARBA/CHANGE_DOC and the /ARBA/LFA1 for the VENDORCREATED, VENDORCHANGED, and VENDORDELETED events.

● The /ARBA/LFA1 is linked to the change document object VENDORCREATED and VENDORCHANGED in the SWEC transaction and has the following field restrictions:
  ○ Name
  ○ Postal Code
  ○ Region
  ○ House or Street Number
  ○ Telephone Number
  ○ Fax Number
  ○ URL
  ○ Vendor Number
  ○ Purchase Organization
  ○ Partner Function
  ○ Center Deletion Flag For Master Data
  ○ Centrally imposed purchasing block
  ○ Function that is blocked

Your SAP ERP maintains the following configuration for the incremental load for suppliers:

In the SW01, the business object, /ARBA/LFAI contains the following events:

● VENDORCREATED
● VENDORCHANGED
● VENDORDELETED

In the SWEC, the change document object event is assigned to the business object /ARBA/LFAI.

The following options are maintained for the create, change, and delete events:

<table>
<thead>
<tr>
<th>Object Type</th>
<th>Event</th>
<th>Select Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/LFAI</td>
<td>VENDORCREATED</td>
<td>On Create</td>
</tr>
<tr>
<td>/ARBA/LFAI</td>
<td>VENDORDELETED</td>
<td>On Change</td>
</tr>
<tr>
<td></td>
<td>VENDORCHANGED</td>
<td>On Change</td>
</tr>
</tbody>
</table>

The following field restrictions are available for the VENDORDELETED and VENDORCHANGED events:

VENDORDELETED
### Change Condition: Field restriction for KRED

<table>
<thead>
<tr>
<th>Not</th>
<th>Express. 1</th>
<th>Operator</th>
<th>Express. 2</th>
<th>And/Or</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;LFA1_LOEVM_OLD&amp;</td>
<td>=</td>
<td></td>
<td></td>
<td>or</td>
</tr>
<tr>
<td>&amp;LFA1_LOEVM_NEW&amp;</td>
<td></td>
<td></td>
<td></td>
<td>and</td>
</tr>
<tr>
<td>&amp;LFA1_SPERM_OLD&amp;</td>
<td>=</td>
<td>X</td>
<td></td>
<td>or</td>
</tr>
<tr>
<td>&amp;LFA1_SPERM_NEW&amp;</td>
<td>=</td>
<td>99</td>
<td></td>
<td>or</td>
</tr>
<tr>
<td>&amp;LFA1_SPERQ_NEW&amp;</td>
<td>=</td>
<td>01</td>
<td></td>
<td>or</td>
</tr>
<tr>
<td>&amp;LFA1_SPERQ_NEW&amp;</td>
<td>=</td>
<td>02</td>
<td></td>
<td>or</td>
</tr>
<tr>
<td>&amp;LFA1_SPERQ_NEW&amp;</td>
<td>=</td>
<td>03</td>
<td></td>
<td>or</td>
</tr>
<tr>
<td>&amp;LFA1_SPERQ_NEW&amp;</td>
<td>=</td>
<td>04</td>
<td></td>
<td>or</td>
</tr>
<tr>
<td>&amp;LFA1_SPERQ_NEW&amp;</td>
<td>=</td>
<td>99</td>
<td></td>
<td>or</td>
</tr>
<tr>
<td>&amp;LFM1_LOEVM_NEW&amp;</td>
<td>=</td>
<td>X</td>
<td></td>
<td>or</td>
</tr>
<tr>
<td>&amp;LFM1_SPERM_NEW&amp;</td>
<td>=</td>
<td>X</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**VENDORCHANGED**
### Change Condition: Field restriction for KRED

<table>
<thead>
<tr>
<th>Field</th>
<th>Operator</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;LFA1_NAME1_OLD&amp;</td>
<td>&lt;&gt;</td>
<td>&amp;LFA1_NAME1_NEW&amp;</td>
</tr>
<tr>
<td>&amp;LFA1_PSTLZ_OLD&amp;</td>
<td>&lt;&gt;</td>
<td>&amp;LFA1_PSTLZ_NEW&amp;</td>
</tr>
<tr>
<td>&amp;LFA1_REGIO_OLD&amp;</td>
<td>&lt;&gt;</td>
<td>&amp;LFA1_REGIO_NEW&amp;</td>
</tr>
<tr>
<td>&amp;LFA1_STRAS_OLD&amp;</td>
<td>&lt;&gt;</td>
<td>&amp;LFA1_STRAS_NEW&amp;</td>
</tr>
<tr>
<td>&amp;LFA1_TELF1_OLD&amp;</td>
<td>&lt;&gt;</td>
<td>&amp;LFA1_TELF1_NEW&amp;</td>
</tr>
<tr>
<td>&amp;LFA1_TELFX_OLD&amp;</td>
<td>&lt;&gt;</td>
<td>&amp;LFA1_TELFX_NEW&amp;</td>
</tr>
<tr>
<td>&amp;LFA1_LFURL_OLD&amp;</td>
<td>&lt;&gt;</td>
<td>&amp;LFA1_LFURL_NEW&amp;</td>
</tr>
<tr>
<td>&amp;WYT3_LIFRN_OLD&amp;</td>
<td>&lt;&gt;</td>
<td>&amp;WYT3_LIFRN_NEW&amp;</td>
</tr>
<tr>
<td>&amp;WYT3_LIFN2_OLD&amp;</td>
<td>&lt;&gt;</td>
<td>&amp;WYT3_LIFN2_NEW&amp;</td>
</tr>
<tr>
<td>&amp;WYT3_EKORG_OLD&amp;</td>
<td>&lt;&gt;</td>
<td>&amp;WYT3_EKORG_NEW&amp;</td>
</tr>
<tr>
<td>&amp;WYT3_PARVW_OLD&amp;</td>
<td>&lt;&gt;</td>
<td>&amp;WYT3_PARVW_NEW&amp;</td>
</tr>
<tr>
<td>&amp;LFA1_LOEVM_OLD&amp;</td>
<td>=</td>
<td>&amp;LFA1_LOEVM_NEW&amp;</td>
</tr>
<tr>
<td>&amp;LFA1_SPERM_OLD&amp;</td>
<td>=</td>
<td>&amp;LFA1_SPERM_NEW&amp;</td>
</tr>
<tr>
<td>&amp;LFA1_SPERQ_OLD&amp;</td>
<td>=</td>
<td>&amp;LFA1_SPERQ_NEW&amp;</td>
</tr>
<tr>
<td>&amp;FM1_LOEVM_OLD&amp;</td>
<td>=</td>
<td>&amp;FM1_LOEVM_NEW&amp;</td>
</tr>
<tr>
<td>&amp;FM1_SPERM_OLD&amp;</td>
<td>=</td>
<td>&amp;FM1_SPERM_NEW&amp;</td>
</tr>
</tbody>
</table>

For email address changes, maintain the **On Change** option and the **VENDORCHANGED** event for the **Change Doc.Obj. ADRESSE**.

### Working with changes to supplier locations and remittance locations

Administrators must make changes when modifying existing supplier locations or remittance locations.

When you make changes to the existing supplier location or remittance location for a supplier and run the supplier incremental load, the changes are not downloaded in the **Supplierlocationconsolidated.csv**, **Remittancelocation.csv**, and **Remittancelocationconsolidated.csv** files.

For example,

**Supplier 1000**

**Supplier Location 2000**
Remittance Location 3000

1. In the Partner Function tab, change the value for the Supplier Location from 2000 to 2100.
2. In the Partner Function tab, change the value for the Remittance Location from 3000 to 3100.
3. Save the changes.

When running the incremental load for suppliers, the newly changed Supplier Location 2100 and Remittance Location 3100 is not available in the SupplierLocationConsolidated.csv and RemittanceLocationConsolidated.csv files.

**Note**
The Ariba Procurement Solution integrated with SAP ERP does not support downloading the changes done for existing supplier locations or remittance locations.

To import the newly changed values for the supplier location and remittance location in the incremental load, administrators must delete the existing supplier location and remittance location and create new values for the supplier location and remittance location for the supplier.

**Note**
When you run the incremental load after deleting the supplier location or remittance location for a supplier, the SupplierLocationConsolidated_Delete.csv files and the RemittanceLocationConsolidated_Delete.csv files display information on the supplier location and remittance location that have been deleted.

### How to make changes to supplier locations and remittance locations

**Context**

To download the changes in the incremental load, administrators must make changes to the supplier locations and remittance locations:

**Procedure**

1. Run the transaction code X1L02 (change mode) for the supplier that needs changes in the supplier location and remittance location.
2. In the Partner Function tab, delete the existing Supplier Location and Remittance Location. Save the changes.
3. In the Partner Function tab, add a new Supplier Location and Remittance Location. Save the changes.

The supplier information displays the newly changed supplier location and remittance location.
Company code import RFC example

The content of the company code import RFC, /ARBA/COMPANY_CODE_EXPORT, provides an example of the mechanisms and techniques:

```plaintext
FUNCTION /ARBA/COMPANY_CODE_EXPORT.
*  "----------------------------- Local Interface: "   *
  " IMPORTING          "   *
  "   VALUE(FILE_NAME) LIKE RLGRAP-FILENAME OPTIONAL
  "   VALUE(ENCODING) TYPE CHAR10 OPTIONAL
  "   VALUE(VARIANT) TYPE /ARBA/TVARV-VARIANT1 OPTIONAL
  "   VALUE(PARTITION) TYPE /ARBA/TVARV-PARTIT OPTIONAL
  " TABLES
  "   COMPANY_INFO STRUCTURE /ARBA/COMPANY_CODE
  "   EXCEPTIONS
  "   NOAUTHORIZATION
  "-----------------------------

  * DATA Declaration
  "-----------------------------

  * Internal table
  "-----------------------------

  types: t_wtab type standard table of /arba/wheretab initial size 0,
  t_select_list type standard table of sdit_qry initial size 0.
  data: wtab type t_wtab,

  * DATA: wtab LIKE /arba/wtab OCCURS 1 WITH HEADER LINE,
  "l_fm_name TYPE rs38l_fnam VALUE '/ARBA/MATERIAL_MANAGE_RFC',
  select_list type t_select_list.

  "-----------------------------

  * Authority check for the Remote Function module
  PERFORM authorization_check USING C_fm_name.
  if sy-subrc <> 0.
    raise noauthorization. "#EC RAISE_OK
  endif.

  * refresh internal table
  REFRESH company_info. CLEAR company_info.
  * Get the addtional filter. The parameter /arba/company_export
  * needs to be maintained in /ARBA/TVARV table
  CALL FUNCTION '/ARBA/PREFILTER'
  EXPORTING
  name = '/ARBA/COMPANY_CODE_EXPORT'
  variant = variant
  partition = partition
  TABLES
  where = wtab
  EXCEPTIONS
  invalid_selection_sign = 1
  invalid_selection_option = 2
  OTHERS = 3.
  IF sy-subrc <> 0. "#EC NEEDED
  ENDIF.

  * Get the Columns to be downloaded
  CALL FUNCTION '/ARBA/SELECT_LIST'
  EXPORTING
  structure_name = '/ARBA/COMPANY_CODE'
  intersection_struct = 'T001'
  TABLES
  select fields = select_list
  EXCEPTIONS
  OTHERS = 1.
  IF sy-subrc <> 0. "#EC NEEDED
  ENDIF.

  * Fetch data's from T001
  SELECT (select_list) FROM t001
  INTO CORRESPONDING FIELDS OF TABLE company_info
```
WHERE (wtab).
*BADI - Modify Company Code
CALL METHOD cl_exithandler=>get_instance
EXPORTING
exit_name = '/ARBA/MASTER_DATA'
null_instance_accepted = 'X'
CHANGING
instance = l_badi
EXCEPTIONS
OTHERS = 1.
IF l_badi IS NOT INITIAL.
CALL METHOD l_badi->publish_companycode
EXPORTING
variant = variant
partition = partition
CHANGING
company_info = company_info[].
ENDIF.
* Downloading the data to the file
IF NOT file_name IS INITIAL.
CALL FUNCTION '/ARBA/DATA_CONVERT_WRITE_FILE'
EXPORTING
i_filename = file_name
i_fileformat = 'CSV'
i_field_separator = ','
i_tablename = '/ARBA/COMPANY_CODE'
i_encoding = encoding
TABLES
i_tab_sender = company_info
EXCEPTIONS
open_failed = 1
close_failed = 2
write_failed = 4
conversion_failed = 5
OTHERS = 6.
IF sy-subrc <> 0. "#EC NEEDED
ENDIF.
ENDIF.
ENDFUNCTION.

Customizing RFC

This section describes how to configure the Ariba Procurement Solution custom RFCs to import additional fields into the Ariba Procurement Solution for existing import integration events. It includes the following sections:

- About extension structures [page 131]
- Adding fields to extension structures [page 132]
- Typical extension structure [page 133]

To customize the Ariba Procurement Solution RFCs, you identify the data to be imported, the RFCs to be used, and the structures where this data appears.
About extension structures

The term extrinsic refers to any data field you import into an Ariba Procurement Solution object that is not part of the object as supplied with the Ariba Procurement Solution. You import extrinsics into the Ariba Procurement Solution from CSV (comma separated value) files, databases, or ERP systems.

Customer packages provide special structures, called extension structures. You use extension structures to define extrinsic data in the Ariba Procurement Solution RFCs on your SAP ERP instance. Most Ariba Procurement Solution RFCs offer one or more extension structures for you to configure extrinsic import tasks. See Summary of Ariba SAP Packages [page 281] for a comprehensive list of customer packages.

Extension structures for import tasks

The Ariba Procurement Solution RFCs use these extension structures to determine which fields to create in the tables they export to the Ariba Procurement Solution. Essentially, extension structures function as a template to define the structure of the Ariba Procurement Solution RFC tables. A diagram of this concept follows:
Extension structures have all the fields in the core Ariba development structures in addition to any fields you add. The contents of an extension structure are shown in the following diagram:

![Diagram of an extension structure]

Extension Structure

Ariba Buyer RFC Structure

- Column 1
- Column 2
- Column 3

Column 4

Column 5

Column 6

This extension structure includes columns 1 through 6

Figure 5: Contents of an Extension Structure

For example, the /ARBA/ASSET_EXPORT extension structure for asset imports contains a substructure called /ARBA/ASSET_EXPORT. The /ARBA/ASSET_EXPORT substructure holds all the base fields that the Ariba Procurement Solution RFCs need to run. To add new asset fields to the Ariba Procurement Solution RFC data imports, you add them to the /ARBA/ASSET_EXPORT structure.

Note

If you change the original structures of Ariba-supplied RFCs, your SAP ERP integration might not function properly. Similarly, you include the original structures as well as the .include statements within the extensions.

This chapter explains only how to define extrinsic data within an extension structure. If you make changes to extension structures, you modify your channel’s data mapping (AML Mapping) to map and transform these fields appropriately. You also modify the Ariba Procurement Solution metadata XML to include your changes.

Adding fields to extension structures

Context

In most circumstances, each structure and extension structure selects data from one table only. Consequently, you return additional fields from one table only.
Procedure

1. Determine the table in SAP ERP that contains the column of data you want to import.
2. Determine which Ariba Procurement Solution RFC selects data from that particular table.
3. Add the field to the RFC extension structure for the table.
4. Click Activate.

Typical extension structure

Assume you want the Ariba Procurement Solution to identify the country where an asset originates. The asset master record, ANLA, contains a column called LAND1, which indicates the country origin of an asset. In this case, the Ariba Procurement Solution RFC that handles asset imports from ANLA is /ARBA/ASSET_EXPORT. For a detailed list of Ariba Procurement Solution SAP objects, see Ariba SAP Objects [page 280].

ARBA/ASSET_EXPORT exports an extension structure called /ARBA/ASSET, which includes the Ariba Procurement Solution structure /ARBA/ASSET. The /ARBA/ASSET structure does not contain the LAND1 field by default, so you add it to the Ariba Procurement Solution extrinsic asset import structure, /ARBA/ASSET, and then activate your changes. Make sure to enter AM_LAND1 for the data element description exactly as it appears in the ANLA table. Subsequently, your SAP ERP includes a new extrinsic, LAND1, in /ARBA/ASSET, which it imports from ANLA.

You use the exact field name and data element descriptions for the column from which you import data. When the Ariba Procurement Solution RFC runs, it uses the information you provided to find the column in the table.

Note

Whenever you make changes to a structure, make sure to click Activate.

Filtering data

Your SAP ERP packages for Ariba Procurement Solution offer three ways to filter, or restrict the data included in a data import. These filtering methods are described in the following sections:

- Static filtering [page 133]
- ABAP filtering [page 138]
- Post-hook filtering [page 140]

Static filtering

Static filtering allows you to construct conditional searches of your SAP ERP tables according to the values of specified fields. The table /ARBA/TVARV allows you to construct queries to limit the data you import from SAP ERP into the Ariba Procurement Solution. Defining queries within the table /ARBA/TVARV is referred to as static...
filtering because the data exists in a structured form within the table /ARBA/TVARV, and data isn’t selected based on dynamically formed queries.

Every Ariba Procurement Solution RFC that imports data from your SAP ERP first makes a call to the table /ARBA/TVARV to see if it needs to restrict its search. The RFCs search for any entries that match their names. If they find a match, they restrict the query according to the instructions in the table /ARBA/TVARV entry. The RFCs used to import master data from SAP ERP into the Ariba Procurement Solution can use the table /ARBA/TVARV to filter the data before it is imported into the Ariba Procurement Solution. Each RFC has a specific table whose fields can be used in table /ARBA/TVARV. For more information, see About /ARBA/TVARV [page 49].

Static filtering is quick and simple, but it does not handle inner join queries against more than one table. If you need to restrict data imports from more than one table, use the filtering techniques explained in ABAP filtering [page 138].

Modifying /ARBA/TVARV

To restrict the data that an Ariba Procurement Solution RFC imports, make a corresponding entry in the table /ARBA/TVARV. Enter a new row in the table /ARBA/TVARV with the RFC name, the field you want to use, and information about what you’re trying to find.

i Note

Ariba suggests you run transaction SM30 to examine the table /ARBA/TVARV. Your SAP user ID must have authorization S_TABU_DIS to run the transaction SM30. The table /ARBA/TVARV resides as a table within the dictionary objects for the /ARBA/MASTER_DATA_EXPORT package.

You also use the table /ARBA/TVARV to configure variable settings in certain RFCs. In particular, you use it to store and retrieve parameters through calls to the /ARBA/TVARV_PARAMETER RFC. This text refers to the table /ARBA/TVARV’s configuration and parameter capabilities as parameter use. To learn more about parameter use, see Configuring parameters [page 49]. For instructions on editing the parameters of the table /ARBA/TVARV, see About /ARBA/TVARV [page 49].

/ARBA/TVARV field reference

The following table presents the fields you set in the table /ARBA/TVARV.

Table 7: /ARBA/TVARV Field Reference (continued)

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Description</th>
<th>Use and Necessity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ariba Variant</td>
<td>Identifies the unique name for the shape of data in your SAP ERP as it appears within the Ariba Procurement Solution. An Ariba Procurement Solution variant is different from a variant in SAP ERP.</td>
<td>Necessary for both filtering and parameter use (selection types S and P).</td>
</tr>
<tr>
<td>Ariba Partition</td>
<td>Describes the unique instance name of a specified variant shape as you represent it to the Ariba Procurement Solution.</td>
<td>Necessary for both filtering and parameter use (selection types S and P).</td>
</tr>
<tr>
<td>Field Name</td>
<td>Description</td>
<td>Use and Necessity</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Variable Name</td>
<td>Provides an index to the table. Ariba Procurement Solution uses the Name field either to pass the</td>
<td>Necessary for both filtering and parameter use (selection types S and P).</td>
</tr>
<tr>
<td></td>
<td>name of the calling function or to pass the name of the parameter it seeks. See Configuring</td>
<td></td>
</tr>
<tr>
<td></td>
<td>parameters [page 49] to find out more about the table /ARBA/TVARV and parameters.</td>
<td></td>
</tr>
<tr>
<td>Field Name</td>
<td>Refers to the column name you use to filter data. Leave this field blank to use the table /ARBA/</td>
<td>Necessary for filtering only (selection type S).</td>
</tr>
<tr>
<td></td>
<td>TVARV for parameters and selection type P.</td>
<td></td>
</tr>
<tr>
<td>Selection Category</td>
<td>Determines whether to use /ARBA/TVARV to filter data or to configure a parameter for an RFC. Enter</td>
<td>Necessary for both filtering and parameter use (selection types S and P).</td>
</tr>
<tr>
<td></td>
<td>a value of S to perform filtering and P to configure parameters.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>See Configuring parameters [page 49] to find out more about the table /ARBA/TVARV and parameters.</td>
<td></td>
</tr>
<tr>
<td>Number</td>
<td>Determines the sequential order of selection statements. The order determines how to construct the</td>
<td>Optional and used for filtering (selection type S).</td>
</tr>
<tr>
<td></td>
<td>entire selection. See Selection condition options [page 136].</td>
<td></td>
</tr>
<tr>
<td>Incl/Excl</td>
<td>Determines whether to include or exclude the selection condition. Set this value to I for inclusion</td>
<td>Required for filtering use (selection type S).</td>
</tr>
<tr>
<td></td>
<td>if the condition is positive or to E for exclusion if the condition is negative.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>See Selection condition options [page 136].</td>
<td></td>
</tr>
<tr>
<td>Option</td>
<td>Allows you to create conditions based on equality, value ranges, and patterns. Set it to the</td>
<td>Required for filtering use (selection type S).</td>
</tr>
<tr>
<td></td>
<td>following conditions:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BT — The “BeTween” option specifies a selection within the range of values in the LOW and HIGH</td>
<td></td>
</tr>
<tr>
<td></td>
<td>fields. For example, names between “Reid” and “Sligh”.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CP — The “Contains Pattern” option allows you to return data with a given pattern, for instance,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>names that begin with “Dan”.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EQ — The “EQual” option determines equivalency, such as names = “Angus”.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>GE — The “Greater than or Equal to” option determines if the FIELDNAME is greater than or equal to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>a value, such as names &gt;= “Bruce”.</td>
<td></td>
</tr>
<tr>
<td>Selection Value</td>
<td>Sets the low value for filtering and settings for parameters. To enter particularly long values—</td>
<td>Optional for parameter configuration and required for most filtering statements.</td>
</tr>
<tr>
<td>(LOW)</td>
<td>like strings—overlap values to the HIGH field.</td>
<td></td>
</tr>
<tr>
<td>Selection Value</td>
<td>Sets the high value and works with LOW values and BT option fields.</td>
<td>Optional for most parameter configuration and filtering, except BT option fields.</td>
</tr>
<tr>
<td>(HIGH)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Selection condition options

The selection options in the /ARBA/TVARV table reflect the internal structure of select options in SAP ERP report tables. The /ARBA/TVARV structure itself resembles the SAP TVARV structure, and the fields function similarly.

The /ARBA/TVARV structure allows you to create the following conditions:

- Equality and inclusion
- Ranges and exclusion
- Patterns and comparisons

Equality and inclusion

Equality and inclusion conditions allow you to set a selection criteria that equals or includes a specified value.

The following example takes you through the steps of creating an equality and inclusion filter condition.

Scenario: You need to create a static filter that pulls only company information for companies in specific countries.

Creating a static filter

1. See /ARBA/COMPANY_CODE_EXPORT to determine where this information originates.

   In this example, the information originates from table T001. Table T001 contains an entry called LAND1, which designates countries. In this example, the only countries you want to import company information from are Australia and Japan, which LAND1 designates with the strings AU and JP.

   You have the information you need to build your static filter:
   - The Ariba Procurement Solution RFC that imports the data
   - The column that contains the data (and the table)
   - The data set you want to return

2. Build your select statement.

   The statement depends on the data set you want to capture. In this example, you want to import rows where LAND1 is equal to AU or rows in which LAND1 is equal to JP. Another way of saying this might be:

   \[(\text{LAND1} = \text{"AU"}) \text{ OR } (\text{LAND1} = \text{"JP"})\]

   In this example, you want to define a logical OR based on two conditions of equality. Each condition in a selection requires its own row in /ARBA/TVARV. The two rows might look similar to the entries in the following table (the VARIANT and PARTITION fields are not included):

<table>
<thead>
<tr>
<th>VARIABLE NAME</th>
<th>FIELD NAME</th>
<th>SELECTION CATEGORY</th>
<th>NUMBER</th>
<th>INCL/EXCL</th>
<th>OPTION</th>
<th>SELECTION VALUE (LOW)</th>
<th>SELECTION VALUE (HIGH)</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/COMPANY_CODE_EXPORT</td>
<td>LAND1</td>
<td>S</td>
<td>0000</td>
<td>I</td>
<td>EQ</td>
<td>AU</td>
<td></td>
</tr>
</tbody>
</table>
The fields in the **VARIABLE NAME** column indicate this filter is for the data imported by the RFC `/ARBA/COMPANY_CODE_EXPORT`.

The fields in the **FIELD NAME** column indicate you are importing data from the `LAND1` column.

The fields in the **SELECTION CATEGORY** column specify to the calling function that the row is for filtering, or selecting, data. In this case, the `S` specifies filtering.

The fields in the **NUMBER** column include more than one row of the same filtering table in the condition. It starts at zero, or 0000. The order is not significant.

The fields in the **INCL/EXCL** column indicate whether to exclude the condition from the statement or include it, signified by `I` or `E`. An included condition is generally specified with an **OR** statement. To specify an excluded condition, you generally use an **AND NOT** statement. For an example of exclusion, see Ranges and exclusion [page 137].

The fields in the **OPTION** column determine that you’re selecting for a condition of equality (**EQ**). It is also possible to specify:

- **Ranges of values** (**BT**)—The “BeTween” option specifies a selection within the range of values in the `LOW` and `HIGH` fields. Such as the names between “Daly” and “Thompson.”

- **Pattern matches** (**CP**)—The “Contains Pattern” option allows you to return data with a given pattern, for example, names that begin with “Lu.” For more information, see Patterns and comparisons [page 138]

- **Greater than or equal to conditions** (**GE**)—The “Greater than or Equal to” option determines if the `FIELDNAME` is greater than or equal to a value, for example names `>= “Bruce.”`

### Ranges and exclusion

You can also create other kinds of selection options. Suppose you want to import only countries whose SAP ERP codes alphabetically range between, and include, AU and GB (for Australia and Great Britain). However, you need to exclude Canada (CA) from this import. Here’s a rough statement of the logic:

```
(“AU” => LAND1 => “GB”) AND NOT (LAND1 = “CA”)
```

The table `/ARBA/TVARV` entries could resemble the rows in the following table.
The selection first includes the range between AU and GB and then excludes CA.

**Patterns and comparisons**

You also select data according to its pattern and by comparing values.

Suppose you want to import only vendors that certain users have created, according to the user's name. You want to import only users who have names that begin with the letter “M” or “N”. For some reason, however, you want only some of the “N” users, specifically, only those in the alphabetic range “N” through “NE”.

The Ariba Procurement Solution RFC, `/ARBA/Vendor_ONLY_EXPORT`, imports vendors from the table `LAFI`. `LAFI` contains the column `ERNAM` for the user name. The logic of the selection can resemble the following statement:

\[(ERNAM \text{ LIKE "M"}) \text{ OR (ERNAM LIKE "N") AND NOT (ERNAM >= "ND")}\]

You do not have to use `AND NOT ERNAM >= "ND"`, but this example does so for illustrative purposes. Using `AND ERNAM < "ND"` is more concise.

The values in `/ARBA/TVARV` might resemble the following:

<table>
<thead>
<tr>
<th>VARIABLE NAME</th>
<th>FIELD NAME</th>
<th>SELECTION TYPE</th>
<th>NUMBER</th>
<th>INCL/EXCL</th>
<th>OPTION</th>
<th>SELECTION VALUE (LOW)</th>
<th>SELECTION VALUE (HIGH)</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/ARBA/Vendor_ONLY_EXPOR T</code></td>
<td>ERNAM</td>
<td>S</td>
<td>0000</td>
<td>I</td>
<td>CP</td>
<td>M</td>
<td></td>
</tr>
<tr>
<td><code>/ARBA/Vendor_ONLY_EXPOR T</code></td>
<td>ERNAM</td>
<td>S</td>
<td>0001</td>
<td>I</td>
<td>CP</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td><code>/ARBA/Vendor_ONLY_EXPOR T</code></td>
<td>ERNAM</td>
<td>S</td>
<td>0002</td>
<td>E</td>
<td>GE</td>
<td>ND</td>
<td></td>
</tr>
</tbody>
</table>

This filter selects three conditions:

- Names that begin with “M.”
- Names that begin with “N.”
- After the first two conditions, the filter refines the set to only those names that are equal to or less than “NE” (or not greater than or equal to “ND”).

**ABAP filtering**

ABAP filtering is a method of selecting data by implementing the BAdI `/ARBA/MASTER_DATA`. To customize programs, use the `PUBLISH_PREFILTER` method to customize the ABAP filtering.

You use conditional tests to determine how and if the BAdI method restricts the data set of the table based on the calling RFC name. This process is called ABAP filtering because you write ABAP code to get the results you want.
This section explains how ABAP filtering:

- Performs inner joins
- Provides an RFC interface
- Works in a sample configuration

## Joins and ABAP filters

You use ABAP filters to reduce your data set through references to more than one table. Because you are writing the code yourself, you can implement inner join statements.

### Prefilter interface

```r
METHOD PUBLISH_PREFILTER.
```

The method parameters are as follows:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>P</th>
<th>O</th>
<th>Typing</th>
<th>Associated Type</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>Importing</td>
<td>Unchecked</td>
<td>Checked</td>
<td>Type</td>
<td>/ARBA/TVARV-NAME</td>
<td>Blank</td>
<td>ABAP: Name of variant variable</td>
</tr>
<tr>
<td>VARIANT</td>
<td>Importing</td>
<td>Unchecked</td>
<td>Checked</td>
<td>Type</td>
<td>/ARBA/TVARV-VARIANT1</td>
<td>Blank</td>
<td>Char20</td>
</tr>
<tr>
<td>PARTITION</td>
<td>Importing</td>
<td>Unchecked</td>
<td>Checked</td>
<td>Type</td>
<td>/ARBA/TVARV-PARTIT</td>
<td>Blank</td>
<td>Char20</td>
</tr>
<tr>
<td>WHERE</td>
<td>Changing</td>
<td>Unchecked</td>
<td>Checked</td>
<td>Type</td>
<td>/ARBA/PREFILTER_T</td>
<td>Blank</td>
<td>Table Type for Prefilter</td>
</tr>
</tbody>
</table>

The RFCs pass your unique Ariba Procurement Solution variant, site, and RFC name.

### Sample ABAP filter

It is possible for you to create dynamic `WHERE` statements for selections in the `WHERE` structure. The example code in `PUBLISH_PREFILTER` demonstrates the use of dynamic `WHERE` statements. The `WHERE` statements in the code can come from another table.

Following is an example of an inner join ABAP filter. This example joins a purchase organization table to a company code table to retrieve all purchase organizations for company codes tied to the US:

```r
TABLES: T001.
DATA: US_BUKRS LIKE T001-BUKRS.
IF '/ARBA/PURCHASE_ORG_EXPORT' = NAME.
  SELECT DISTINCT BUKRS INTO US_BUKRS FROM T001
  WHERE LAND1 = 'US'.
  DESCRIBE TABLE WHERE LINES COUNT.
```
IF COUNT > 0.  
  WHERE = 'or'. APPEND WHERE.  
ENDIF.  
CONCATENATE 'bukrs = ''' US_BUKRS '''' INTO WHERE.  
APPEND WHERE.  
ENDSELECT.  
ENDIF.

### Post-hook filtering

Both static filters and ABAP filters cut down on data before an Ariba Procurement Solution RFC begins to process anything. You filter data after the Ariba Procurement Solution RFCs complete through post-hook filtering. SAP ERP commonly provides this feature through BAdI methods.

This section introduces:

- BAdI methods
- Post-hook filtering suggestions

### BAdI methods

The available BAdI methods are as follows:

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUBLISH_COMPANYCODE</td>
<td>Publish Company Code Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_PLANT</td>
<td>Publish Plant Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_PREFILTER</td>
<td>Publish Prefilter Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_PURCHASE_GROUP</td>
<td>Publish Purchase Group Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_PURCHASE_ORGANIZATION</td>
<td>Publish Purchase Organization Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_ACCOUNT_CATEGORY</td>
<td>Publish Account Category Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_ASSET</td>
<td>Publish Asset Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_COST_CENTER</td>
<td>Publish Cost Center Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_GENERAL_LEDGER</td>
<td>Publish General Ledger Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_INTERNAL_ORDER</td>
<td>Publish Internal Order Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_WBS</td>
<td>Publish WBS Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_MATERIAL_GROUP</td>
<td>Publish Material Group Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_CURRENCY_CONVERSION</td>
<td>Publish Currency Conversion Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_PLANT_PURCHASE_ORG</td>
<td>Publish Plant Purchase Organization Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_ACCOUNT_FIELD</td>
<td>Publish Account Field Details After Processing</td>
</tr>
<tr>
<td>PUBLISH_VENDOR_ONLY</td>
<td>Publish Vendor Only Details After Processing</td>
</tr>
</tbody>
</table>
The Ariba Procurement Solution provides post-hook filtering through BAdI methods. The Ariba Procurement Solution RFCs call BAdI methods just before they exit.

Each Ariba Procurement Solution RFC has its own BAdI method. Here is an example for the `PUBLISH_ASSET`:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Part Value</th>
<th>Optional</th>
<th>Typing Method</th>
<th>Associated Type</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VARIANT</td>
<td>Importing</td>
<td>Unchecked</td>
<td>Checked</td>
<td>Type</td>
<td>/ARBA/TVARV-VARIANT1</td>
<td>Blank</td>
<td>Char20</td>
</tr>
<tr>
<td>PARTITION</td>
<td>Importing</td>
<td>Unchecked</td>
<td>Checked</td>
<td>Type</td>
<td>/ARBA/TVARV-PARTIT</td>
<td>Blank</td>
<td>Char20</td>
</tr>
<tr>
<td>ASSET_INFO</td>
<td>Changing</td>
<td>Unchecked</td>
<td>Checked</td>
<td>Type</td>
<td>/ARBA/ASSET_T</td>
<td>Blank</td>
<td>Table Type for Prefilter</td>
</tr>
</tbody>
</table>

### Post-hook filtering suggestions

The Ariba Procurement Solution RFCs expose filtering for only some of the tables from which they take data in SAP ERP. The Ariba Procurement Solution RFCs sometimes depend on filtering for one table to restrict the data imported from a second table. For example, you might filter one table for a data type in the first table, and then use that filtered result to import data from a second table.

In most circumstances, this arrangement works just fine. However, if the Ariba Procurement Solution RFCs retrieve data from the second table that you do not want, you need some mechanism to restrict it further. ABAP
and static filters do not work because they only provide you access to the first table, not the second. You need to restrict the data after the Ariba Procurement Solution RFCs have finished with it. Post-hook filters let you restrict the data.

You need to write ABAP code to change the data within the given internal table for an Ariba Procurement Solution RFC.

### Master data integration events

Ariba Procurement Solution imports master data from CSV files that the SAP integration program `/ARBA/MASTER_DATA_EXPORT` creates on your SAP instance client.

This section has the following topics:

- Understanding the Master Data Import [page 142]
- Exporting Master Data Using the User Interface [page 142]
- Running the Master Data Import Event [page 145]
- Checking Output CSV Files [page 147]
- About Importing Data in Multiple Languages [page 149]

### Understanding the master data import

The following high-level steps describe the master data import process:

1. Run the SAP program `/ARBA/MASTER_DATA_EXPORT`. This program transfers the SAP master data into Ariba CSV files on the SAP server.
2. Set up the Ariba Data Transfer Tool on the SAP server to run when the master data download is complete.
3. The Ariba Data Transfer Tool picks up the master data CSV files and sends them to the Ariba Procurement Solution.
4. The Ariba Procurement Solution then creates the master data objects.

**Note**

- All master data loads can be run as full loads or incremental master data loads
- The `/ARBA/MASTER_DATA_EXPORT` program displays the text for the fields only in English. Ensure that you log in to the SAP ERP system in the English (EN) language.

### Integrating master data using the user interface

The `/ARBA/MASTER_DATA_EXPORT` program has an user interface that provides different options to choose either a full or incremental load while running the master data events using the Ariba Integration Toolkit. You can use the Ariba Integration Toolkit option to transfer master data.
The following options are available while running the `ARBA/MASTER_DATA_EXPORT` program using the Ariba Integration Toolkit:

<table>
<thead>
<tr>
<th>Option</th>
<th>This option allows you to...</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Connectivity</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Ariba Integration Toolkit (ITK)</strong></td>
<td>Use the Ariba Integration Toolkit to run the master data integration events. If you do not use the Ariba Integration Toolkit, but use your own tool, choose this option and specify the export tasks to transfer master data.</td>
</tr>
<tr>
<td><strong>Load Option</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Full Load</strong></td>
<td>Import the complete master data from your SAP ERP to your Ariba Procurement Solution. The <strong>Full Load</strong> option allows you to run all the import master data tasks. This option uses the <strong>Load</strong> and <strong>Delete</strong> operations.</td>
</tr>
<tr>
<td><strong>Incremental Load</strong></td>
<td>Import only the changed or newly added master data from your SAP ERP to your Ariba Procurement Solution. The <strong>Incremental Load</strong> option allows you to run the following import master data tasks:</td>
</tr>
<tr>
<td></td>
<td><strong>Accounting Data:</strong></td>
</tr>
<tr>
<td></td>
<td>● <strong>GeneralLedger</strong></td>
</tr>
<tr>
<td></td>
<td>● <strong>InternalOrder</strong></td>
</tr>
<tr>
<td></td>
<td>● <strong>CostCenter</strong></td>
</tr>
<tr>
<td></td>
<td>● <strong>WBSElement</strong></td>
</tr>
<tr>
<td></td>
<td>● <strong>Asset</strong></td>
</tr>
<tr>
<td></td>
<td>● <strong>Supplier Related Data:</strong></td>
</tr>
<tr>
<td></td>
<td>● <strong>Supplier</strong></td>
</tr>
<tr>
<td></td>
<td>● <strong>User Related Data:</strong></td>
</tr>
<tr>
<td></td>
<td>● <strong>User and User Group</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Funds Management Data</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Earmarked Fund</strong></td>
</tr>
<tr>
<td></td>
<td>The incremental change transaction for the earmarked fund supports the following:</td>
</tr>
<tr>
<td></td>
<td>● Creating a new earmarked fund</td>
</tr>
<tr>
<td></td>
<td>● Changing an existing earmarked fund</td>
</tr>
<tr>
<td></td>
<td>● Marking an existing line item for deletion</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>You must perform a one-time initial setup of <code>/ARBA/INCR_DTTM</code>. Incremental <strong>Earmarked Fund</strong> data import from SAP ERP is based only on the date and not the time.</td>
</tr>
<tr>
<td><strong>Scope of Ariba Solution</strong></td>
<td></td>
</tr>
<tr>
<td><strong>P2P Customers</strong></td>
<td>Import master data for Ariba Procure-to-Pay customers.</td>
</tr>
<tr>
<td><strong>P2O Customers</strong></td>
<td>Import master data for Ariba Procure-to-Order customers. This option does not support downloading the remittance location of suppliers.</td>
</tr>
<tr>
<td>Option</td>
<td>This option allows you to...</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Configuration</strong></td>
<td></td>
</tr>
<tr>
<td>Encoding to be used</td>
<td>Specify the encoding type for the data transfer or use the default value available to you.</td>
</tr>
<tr>
<td>Variant</td>
<td>Specify the unique variant name for the Ariba Procurement Solution in your SAP ERP for purchase orders. For example, VREALM_1. If you have maintained a value for the variant in the /ARBA/TVARV table, ensure that you maintain the same value in this field. If the values differ, an error occurs and the files fail during download.</td>
</tr>
<tr>
<td>Partition</td>
<td>Specify the unique name of your site for the Ariba Procurement Solution in your SAP ERP for purchase orders. For example, PREALM_1. If you have maintained a value for the partition in the table /ARBA/TVARV, ensure that you maintain the same value in this field. If the values differ, an error occurs and the files fail during download.</td>
</tr>
<tr>
<td>Download File Location</td>
<td></td>
</tr>
<tr>
<td>Dir path for file download</td>
<td>Specify the directory path for downloading the files.</td>
</tr>
<tr>
<td><strong>Choose Master Data for Import</strong></td>
<td></td>
</tr>
<tr>
<td>Organization Data</td>
<td>Import data using the following options:</td>
</tr>
<tr>
<td></td>
<td>● Plant</td>
</tr>
<tr>
<td></td>
<td>● CompanyCode</td>
</tr>
<tr>
<td></td>
<td>● PurchaseOrg</td>
</tr>
<tr>
<td></td>
<td>● PlantPurchaseOrg</td>
</tr>
<tr>
<td></td>
<td>● PurchaseGroup</td>
</tr>
<tr>
<td>Accounting Data</td>
<td>Import data using the following options:</td>
</tr>
<tr>
<td></td>
<td>● GeneralLedger</td>
</tr>
<tr>
<td></td>
<td>● InternalOrder</td>
</tr>
<tr>
<td></td>
<td>● CostCenter</td>
</tr>
<tr>
<td></td>
<td>● WBSElement</td>
</tr>
<tr>
<td></td>
<td>● Asset</td>
</tr>
<tr>
<td>Cross/Application Configuration Data</td>
<td>Import data using the following options:</td>
</tr>
<tr>
<td></td>
<td>● AccountCategory</td>
</tr>
<tr>
<td></td>
<td>● AcctCategoryFieldStatus</td>
</tr>
<tr>
<td></td>
<td>● ERPCommodityCode</td>
</tr>
<tr>
<td></td>
<td>● CurrencyConversion</td>
</tr>
<tr>
<td></td>
<td>● TaxCode</td>
</tr>
<tr>
<td></td>
<td>● Payment Terms</td>
</tr>
</tbody>
</table>
You can use the `/ARBA/MASTER_DATA` Business AddIns (BADI) to customize the master data for the supplier, user, and payment terms.

### Related Information

**About importing data in multiple languages** [page 149]

### How to run the master data import event

**Context**

The input for the program `/ARBA/MASTER_DATA_EXPORT` includes the directory where the CSV files should be written, encoding, and the data to import.

**Procedure**

1. In the command field of the SAP ERP main screen, enter `/N/ARBA/OND_PROC`.
2. Navigate to `SAP menu > Master Data > Report` option to expand the tree structure.
3. Double-click **Master Data Report Integration**. The **Export All Master Data Required For Ariba On-Demand Application** page appears.

4. Do one of the following:
   - Click Ariba Integration Toolkit to run the master data using the Ariba Integration Toolkit or your own tool.
   - Click **Direct Connectivity** to run the master data tasks directly. For more information, see Integrating Master Data Directly From an SAP ERP System [page 152].

5. Choose one of the following load options:
   - Click **Full Load** to run the complete master data on your SAP ERP system.
   - Click **Incremental Load** to fetch the changed or updated master data on your SAP ERP system.

6. Choose one of the Ariba solutions in the **Scope of Ariba Solution** section.

7. Enter the following configuration for the master data:
   - **Encoding to be used**: Specify the encoding type or use the default value available. For example, UTF-8.
   - **Variant**: Specify the variant for your Ariba Procurement Solution.
   - **Partition**: Specify the partition for your Ariba Procurement Solution.

   **Note**
   If you have maintained values for the variant and partition in the table /ARBA/TVARV, ensure that you maintain the same values in these fields. If the values differ, an error occurs and files are not downloaded.

8. Specify the file path to the application server to download files in the **Download File Location** section.

   **Note**
   This option is only available if you have checked Ariba Integration Toolkit.

9. Choose the master data tasks to be exported to the Ariba Procurement Solution.

10. Click the **Execute** button.

**Next Steps**

- To schedule /ARBA/MASTER_DATA_EXPORT to run at regular intervals, save the program input as a variant that you can specify to run as a job. Schedule this program using the transaction code SM36. For more information about this transaction or scheduling SAP function modules, see your SAP documentation.
- When writing the master data CSV files is complete:
  - Schedule the Ariba Data Transfer Tool to run the event **Import Batch Data** to load the CSV files each time /ARBA/MASTER_DATA_EXPORT has written them.
  - Make sure the Ariba Data Transfer Tool input directory specification matches the directory where you created the CSV files, as noted in Step 2.
## Checking output CSV files

When the program `:/ARBA/MASTER_DATA_EXPORT` has run successfully, you can check that the CSV files are generated as expected and that they are located where the Ariba Data Transfer Tool can find them.

The program `:/ARBA/MASTER_DATA_EXPORT` creates a temporary directory called `processingyyyymmddhhmmss`, where `yyyymmddhhmmss` is the date and time stamp indicating when the directory was created. It writes all the requested CSV files into this directory, depending on which data you chose to pull. The `Processing` directory is placed inside the folder you specified as input in Step 2, earlier.

### Note

Set your Ariba Data Transfer Tool options file to specify `processing` as the `filterPrefix` value so that the Ariba Data Transfer Tool ignores it.

When the program `:/ARBA/MASTER_DATA_EXPORT` is finished writing the files, it renames the temporary directory to `processedyyyymmddhhmmss`. This directory is where the Ariba Data Transfer Tool looks for the CSV files. The next time the Ariba Data Transfer Tool runs, it finds this directory and transfers the data.

The following table lists the SAP remote function calls (RFCs) and the CSV files they generate for each data import task.

<table>
<thead>
<tr>
<th>RFC Used</th>
<th>Generated CSV File</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/ARBA/COMPANY_CODE_EXPORT</code></td>
<td>CompanyCode.csv</td>
</tr>
<tr>
<td><code>/ARBA/PLANT_EXPORT</code></td>
<td>Plant.csv</td>
</tr>
<tr>
<td><code>/ARBA/PURCHASE_ORG_EXPORT</code></td>
<td>PurchaseOrg.csv</td>
</tr>
<tr>
<td><code>/ARBA/PURCHASE_GROUP_EXPORT</code></td>
<td>PurchaseGroup.csv</td>
</tr>
<tr>
<td><code>/ARBA/PLANT_PURCHASE_EXPORT</code></td>
<td>PlantPurchaseOrgCombo.csv</td>
</tr>
<tr>
<td><code>/ARBA/ASSET_EXPORT</code></td>
<td>Asset.csv</td>
</tr>
<tr>
<td><code>/ARBA/COST_CENTER_EXPORT</code></td>
<td>CostCenter.csv</td>
</tr>
<tr>
<td><code>/ARBA/GENERAL_LEDGER_EXPORT</code></td>
<td>GeneralLedger.csv</td>
</tr>
<tr>
<td><code>/ARBA/INTERNAL_ORDER_EXPORT</code></td>
<td>InternalOrder.csv</td>
</tr>
<tr>
<td><code>/ARBA/WBS_EXPORT</code></td>
<td>WBSElement.csv</td>
</tr>
<tr>
<td><code>/ARBA/ACCOUNT_CATEGORY_EXPORT</code></td>
<td>AccountCategory.csv</td>
</tr>
<tr>
<td><code>/ARBA/ACCOUNT_FIELD_EXPORT</code></td>
<td>AccCategoryFieldStatus Combo.csv</td>
</tr>
<tr>
<td><code>/ARBA/INTERNAL_ORDER_EXPORT</code></td>
<td>CompanyCodeIOCombo.csv</td>
</tr>
<tr>
<td><code>/ARBA/WBS_EXPORT</code></td>
<td>CompanyCodeWBSCombo.csv</td>
</tr>
<tr>
<td><code>/ARBA/MATERIAL_GROUP_EXPORT</code></td>
<td>ERPCommodityCode.csv</td>
</tr>
<tr>
<td><code>/ARBA/CURRENCY_CONVERT_EXPORT</code></td>
<td>CurrencyConversion_Rate.csv</td>
</tr>
<tr>
<td>RFC Used</td>
<td>Generated CSV File</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>---------------------------------------------------------</td>
</tr>
<tr>
<td>/ARBA/VENDOR_EXPORT</td>
<td>PurchaseOrgSupplierCombo.csv</td>
</tr>
<tr>
<td></td>
<td>SupplierConsolidated.csv</td>
</tr>
<tr>
<td></td>
<td>SupplierLocationConsolidated.csv</td>
</tr>
<tr>
<td></td>
<td>RemittanceLocationConsolidated.csv</td>
</tr>
<tr>
<td>/ARBA/SUPPLIER_LOCATION_EXPORT</td>
<td>SupplierLocationConsolidated.csv</td>
</tr>
<tr>
<td>/ARBA/PAYMENT_TERM_EXPORT</td>
<td>PaymentTermsConsolidated.csv</td>
</tr>
<tr>
<td>/ARBA/REMITTANCE_LOCATION_EXPORT</td>
<td>RemittanceLocationConsolidatedExport.csv</td>
</tr>
<tr>
<td>/ARBA/USER_EXPORT</td>
<td>UserConsolidated.csv</td>
</tr>
<tr>
<td>/ARBA/USER_GROUP_EXPORT</td>
<td>GroupConsolidated.csv</td>
</tr>
<tr>
<td>/ARBA/TAX_CODE_EXPORT</td>
<td>TaxCode.csv</td>
</tr>
</tbody>
</table>

The following table lists the class used and the CSV files they generate for each Funds Management data import task.

<table>
<thead>
<tr>
<th>Class Used</th>
<th>Generated CSV File</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_BP_EXPORT</td>
<td>BudgetPeriod.csv</td>
</tr>
<tr>
<td>/ARBA/CL_COMMIT_EXPORT</td>
<td>Commit.csv</td>
</tr>
<tr>
<td>/ARBA/CL_FMAREA_EXPORT</td>
<td>FMArea.csv</td>
</tr>
<tr>
<td>/ARBA/CL_GRANT_EXPORT</td>
<td>Grantee.csv</td>
</tr>
<tr>
<td>/ARBA/CL_FUNCTAREA_EXPORT</td>
<td>FunctionalArea.csv</td>
</tr>
<tr>
<td>/ARBA/CL_FUNDS_CTR_EXPORT</td>
<td>FundsCenter.csv</td>
</tr>
<tr>
<td>/ARBA/CL_FUND_EXPORT</td>
<td>Fund.csv</td>
</tr>
<tr>
<td>/ARBA/CL_EARMARKED_EXPORT</td>
<td>● EarmarkedFundsDocument.csv</td>
</tr>
<tr>
<td></td>
<td>● EarmarkedFundsItem.csv</td>
</tr>
</tbody>
</table>

Buyers can run the transaction code SLG1 to view all messages in the application log in SAP ERP.

**How to view the application log**

**Procedure**

1. Go to TCode SLG1. The Analyze Application Log appears.
2. Enter /ARBA/MASTERDATA in the **Object** field.
3. Enter /ARBA/SUB_MASTERDATA in the **SubObject** field.
About importing data in multiple languages

The /ARBA/MASTER_DATA_LANG_EXPORT program provides options to download translations of the following master data:

<table>
<thead>
<tr>
<th>RFC Used</th>
<th>Generated CSV File</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/GENRAL_LDGR_NAMES_EXPORT</td>
<td>GeneralLedgerLang.csv</td>
</tr>
<tr>
<td>/ARBA/ACCOUNT_CAT_NAMES_EXPORT</td>
<td>AccountCategoryLang.csv</td>
</tr>
<tr>
<td>/ARBA/PAYMENT_TERM_NAMES_EXPORT</td>
<td>PaymentTermsLang.csv</td>
</tr>
<tr>
<td>/ARBA/TAX_CODE_NAMES_EXPORT</td>
<td>TaxCodeLang.csv</td>
</tr>
<tr>
<td>/ARBA/MATERAL_GRP_NAMES_EXPORT</td>
<td>CommodityCodeLang.csv</td>
</tr>
<tr>
<td>/ARBA/COST_CENTER_NAMES_EXPORT</td>
<td>CostCenterLang.csv</td>
</tr>
</tbody>
</table>

Table 8: Funds Management Master Data

<table>
<thead>
<tr>
<th>Class Used</th>
<th>Generated CSV File</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_BP_LANG_EXPORT</td>
<td>BudgetPeriod.csv</td>
</tr>
<tr>
<td>/ARBA/CL_COMMIT_LANG_EXPORT</td>
<td>Commit.csv</td>
</tr>
<tr>
<td>/ARBA/CL_FMAREA_LANG_EXPORT</td>
<td>FMArea.csv</td>
</tr>
<tr>
<td>/ARBA/CL_GRANT_LANG_EXPORT</td>
<td>Grantee.csv</td>
</tr>
<tr>
<td>/ARBA/CL_FUNCTAREA_LANG_EXPORT</td>
<td>FunctionalArea.csv</td>
</tr>
<tr>
<td>/ARBA/CL_FUNDS_CTR_LANG_EXPORT</td>
<td>FundsCenter.csv</td>
</tr>
<tr>
<td>/ARBA/CL_FUND_LANG_EXPORT</td>
<td>Fund.csv</td>
</tr>
</tbody>
</table>

You can run the following Master Data Language Export tasks directly or using the Ariba Integration Toolkit using the program /ARBA/MASTER_DATA_LANG_EXPORT:

- Account Category Language
- General Ledger Language
- TaxCode Language
- Cost Center Language
- ERPCommodityCode Language
- Payment Terms Language
- Funds Management Data Language

How to import data in multiple languages from SAP ERP

Procedure

1. Download the relevant SAP transports from https://connect.ariba.com. In the Readme file, the SAP transport corresponding to the description Contains objects for Master data pulls and Language pulls (File) must be imported into your SAP ERP system.
2. Configure the Import ERP Master Data Translations data import task for use with the Ariba Data Transfer Tool. For information on configuring data import tasks in the Data Transfer Tool, refer to the Ariba Integration Toolkit Guide.

Customizing Funds Management extracts

You can customize the Funds Management extract from SAP ERP to Ariba Procurement Solution to meet your requirements.

To add a new field to extract Funds Management data from SAP ERP and map the field to Ariba Procurement Solution, see the following sections:

How to add custom fields for Funds Management extract in SAP ERP

Prerequisites

Identify the export for which you want to perform the customization.

Context

You can customize extraction of Funds Management data to extract only the Funds Management data relevant for you. The following procedure lists the steps required to add field(s) to extract Funds Management data:

Procedure

1. Copy the class name. For example, /ARBA/CL_FUND_EXPORT.
2. From the SAP ERP main screen, navigate to Tools > ABAP Workbench > Development > SAP Business Workflow > Definition tools > Application Integration > Class Builder.
   
   Class Builder: Initial Screen window appears.
3. Paste the class name in the Object type field.
4. Click Display.
   
   Class Builder: Display Class /ARBA/CL_FUND_EXPORT window appears which lists all the methods.
5. Double-click GET_STRUCT_NAME method listed under Methods.
6. Copy the STRUCTURE_NAME, /ARBA/FMFINCODE_STR.
7. From the SAP ERP main screen, navigate to Tools > ABAP Workbench > Development > ABAP Dictionary.
ABAP Dictionary: Initial Screen window appears.
8. Paste the structure name /ARBA/ FMFINCODE_STR in the Data Type field.
9. Click Display.

Dictionary: Initial Screen screen appears.
10. To add a new field, click Display/Change icon (Ctrl+F1).
11. Add a new field, for example, FINUSE by entering the following details and then press Enter:
   - Component: FINUSE
   - Typing Method: Types
   - Component Type: FM_FINUSE (data element name)

   Data Type, Length, Decimal, and Short Description fields are auto-populated.
12. To add more fields in the structure name, repeat the above steps and then click Save.

   Prompt for local Workbench Request appears. Close this prompt.
13. Click Activate (Ctrl+F3).
14. Repeat Step 6 through Step 13 to activate the STRUCTURE_NAME, CSV_STRUCT_Name.

Next Steps

You must map the custom fields for Funds Management extract added in SAP ERP to Ariba Procurement Solution.

How to map custom fields for Funds Management extract between SAP ERP and the Ariba Procurement Solution

Context

The names of the custom fields for Funds Management extract in SAP ERP may be different from the names of these fields in the Ariba Procurement Solution. To map these custom fields, perform the following steps in SAP ERP.

Procedure

1. In the command field of the SAP ERP main screen, type /N/ARBA/OND_PROC.
   SAP Easy Access Transaction Code for Ondemand Procurement window appears.
2. Navigate to SAP menu ➔ Configuration Parameters ➔ Define SAP to Buyer field Map ➔ Display View "Table to maintain mapping of SAP to Buyer Fields": Overview window appears.
3. To add a new field, click Display/Change icon (Ctrl+F1).
Caution: The table is cross-client window appears. Click Continue to continue.

4. Click New Entries (F5).

New Entries: Overview of Added Entries window appears.

5. Add a new field to map from SAP ERP to Ariba Procurement Solution, for example, FINUSE by entering the following details:
   ○ Structure: /ARBA/FUND_STR (CSV structure in SAP ERP)
   ○ Field Sap: FINUSE (field name in SAP ERP)
   ○ Field Ariba: Source Fund (field name in Ariba Procurement Solution)

6. Click Save (Ctrl+S).

Prompt for Workbench request window appears. Click Continue to continue.

7. In the Create Task window, click Continue.

The FINUSE field in SAP ERP is now mapped to the Source Fund field in Ariba Procurement Solution.

To map additional fields, as needed, repeat Step 4 through Step 7.

Integrating master data using the direct connectivity integration method from SAP ERP

This section has the following topics:

- Limitations of Integrating Master Data Directly from an SAP ERP System [page 153]
- Exporting Master Data Directly Using the User Interface [page 153]

Integrate master data directly from SAP ERP to the Ariba Procurement Solution seamlessly without using the Ariba Integration Toolkit through the Direct Connectivity Integration method.

Note

This feature is supported on SAP ERP 6.0 SPS02 (SAP_APPL 600 SPS02 and SAP_BASIS 700 SP18) and higher versions.

Buyers must configure the necessary parameters required to export master data seamlessly. You must:

- Configure the LPCONFIG if you are using ECC6.0 EHP3
- Configure the SOAMANAGER if you are using ECC 6.0 EHP4 and higher

Buyers can run the transaction code SLG1 to view all messages in the application log in SAP ERP. For more information, see Viewing the Application Log [page 148].

Prerequisites

- You must maintain the parameter /ARBA/TEMP_DIRECTORY in the /ARBA/TVARV table before you export the master data through the Direct Connectivity Integration method. For more information see, Maintaining a Temporary Directory While Running Master Data Directly [page 69].
- Before you export the master data, you must create the Authorization Object class. For more information, see Defining authorizations for running Ariba Procure-to-Pay transactions [page 90].
• Ensure that you maintain the following parameters on your SAP ERP:
  Run the transaction code /nsm30, go to the table /ARBA/AUTH_PARAM, and enter the following values:
  ○ **Realm**: Specify the name of your company as defined in your Ariba Procurement Solution. If you use something like, `http://mycompany-T.procurement.ariba.com` to access your Ariba product, specify your company name as `mycompany-T`.
  ○ **Shared Secret**: Specify the shared secret in the SOAMANAGER or LPCONFIG based on your SAP ERP EHP version.
  ○ **Wait (in minutes)**: Specify the wait duration between the retry attempts.
• Ensure the you install the client certificates before running the master data.
  ○ While you export master data, the Ariba Procurement Solution attempts to connect to the procurement system up to three times.
  ○ If the export master data fails due to incorrect shared secret, go to the table /ARBA/AUTH_PARAM, enter the correct shared secret and export the master data again.
• Create a logical file path for the `ISMW_ROOT`. For more information, see How to create a logical file path [page 158].

Limitations when integrating master data directly from SAP ERP [page 153]
Importing master data using the direct connectivity integration method [page 153]

Related Information

Direct connectivity integration [page 16]
Integrating master data using the mediated connectivity integration method [page 161]

Limitations when integrating master data directly from SAP ERP

• When you run the master data tasks without using the Ariba Integration Toolkit, you cannot do the following:
  ○ Use the NTLM and digest proxyAuthMod authentication methods while exporting master data.
  ○ Send any email notifications when the mater data integration fails. Buyers can run the transaction code SLG1 to view all messages in the application log in SAP ERP.
  ○ Run the Incremental Delete for the User Information master data.
  ○ Run the Incremental Load for the Plant Export master data.

Importing master data using the direct connectivity integration method

The program ARBA/MASTER_DATA_EXPORT has an user interface that provides different options to choose either a full or an incremental load while running the master data events directly. You can use the option Direct Connectivity to transfer mater data.
The following options are available while running the program `ARBA/MASTER_DATA_EXPORT` using Direct Connectivity:

<table>
<thead>
<tr>
<th>Option</th>
<th>This option allows you to....</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Connectivity</strong></td>
<td></td>
</tr>
<tr>
<td>Direct Connectivity</td>
<td>Run the master data integration events without the use of the Ariba Integration Toolkit. The master data is sent directly to the Ariba Procurement Solution using the function module <code>/ARBA/DIRECT_CONNECT</code>.</td>
</tr>
<tr>
<td><strong>Load Option</strong></td>
<td></td>
</tr>
<tr>
<td>Full Load</td>
<td>Import the complete master data from your SAP ERP to your Ariba Procurement Solution. This option uses the Load and Delete operations. The <strong>Full Load</strong> option allows you to run all the import master data tasks.</td>
</tr>
<tr>
<td>Incremental Load</td>
<td>Import only the changed or newly added master data from your SAP ERP to your Ariba Procurement Solution. The incremental load option uses the Load operation. The <strong>Incremental Load</strong> option allows you to run the following export master data tasks:</td>
</tr>
<tr>
<td></td>
<td><strong>Accounting Data:</strong></td>
</tr>
<tr>
<td></td>
<td>● GeneralLedger</td>
</tr>
<tr>
<td></td>
<td>● InternalOrder</td>
</tr>
<tr>
<td></td>
<td>● CostCenter</td>
</tr>
<tr>
<td></td>
<td>● WBSElement</td>
</tr>
<tr>
<td></td>
<td>● Asset</td>
</tr>
<tr>
<td></td>
<td>Supplier Related Data:</td>
</tr>
<tr>
<td></td>
<td>● Supplier</td>
</tr>
<tr>
<td></td>
<td>● Supplier Location</td>
</tr>
<tr>
<td></td>
<td>● Remittance Location</td>
</tr>
<tr>
<td></td>
<td>● PurchaseOrgSupplierCombo</td>
</tr>
<tr>
<td></td>
<td>User Related Data:</td>
</tr>
<tr>
<td></td>
<td>● User and User Group</td>
</tr>
<tr>
<td><strong>Scope of Ariba Solution</strong></td>
<td>Import master data for Ariba Procure-to-Pay customers.</td>
</tr>
<tr>
<td>P2P Customers</td>
<td>Import master data for Ariba Procure-to-Order customers. This option does not support downloading the remittance location of suppliers.</td>
</tr>
<tr>
<td><strong>Configuration</strong></td>
<td>Specify the encoding type for the data transfer or use the default value available to you.</td>
</tr>
<tr>
<td>Encoding to be used</td>
<td>Specify the unique variant name for the Ariba Procurement Solution in your SAP ERP for purchase orders. For example, <code>VREALM_1</code>. If you have maintained a value for the variant in the table <code>/ARBA/TVARV</code>, ensure that you maintain the same value in this field. If the values differ, an error occurs and the files fail during download.</td>
</tr>
<tr>
<td>Option</td>
<td>This option allows you to....</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Partition</td>
<td>Specify the unique name of your site for the Ariba Procurement Solution in your SAP ERP for purchase orders. For example, PREALM_1 If you have maintained a value for the partition in the table /ARBA/TVARV, ensure that you maintain the same value in this field. If the values differ, an error occurs and the files fail during download.</td>
</tr>
<tr>
<td>Choose Master Data for Export</td>
<td>Import data using the following options:</td>
</tr>
</tbody>
</table>
| Organization Data             | ● Plant  
● CompanyCode  
● PurchaseOrg  
● PlantPurchaseOrg  
● PurchaseGroup                                                                                                                                               |
| Accounting Data               | Import data using the following options:                                                                                                                                                                                   |
| ● GeneralLedger               | ● InternalOrder  
● CostCenter  
● WBSElement  
● Asset                                                                                                                                                       |
| Cross/Application Configuration Data | Import data using the following options:                                                                                                                                                                                      |
| ● AccountCategory             | ● AcctCategoryFieldStatus  
● ERPCommodityCode  
● CurrencyConversion  
● TaxCode  
● Payment  
● Terms                                                                                                                                                                      |
| Supplier Related Data         | Import supplier data using the Supplier option.                                                                                                                                                                               |
| User Data                     | Import data for users and user groups using the User & User Group option.                                                                                                                                                     |

This section has the following procedures:

- How to configure SOAMANAGER [page 156]
- How to configure LPCONFIG for ECC6.0 EHP3 [page 157]
- Maintaining table entries [page 105]
- How to install the SSL client certificate for HTTPS connection [page 159]
- How to configure the client certificate for certificate-based authentication [page 161]
- How to create a logical file path [page 158]
How to configure SOAMANAGER

**Context**

You can use the SOAMANAGER configuration to set up Direct Connectivity between Ariba Procurement Solution and SAP ERP.

**Procedure**

1. Run the transaction code **SOAMANAGER**.
2. In the Service Administration tab, click **Web Service Configuration**.
3. Specify the following **Search Criteria**:
   - In the **Search By** dropdown, choose **Consumer Proxy**.
   - Choose **Contains** from the options.
   - Enter the first few characters of the proxy name, /ARBA/CO_MIOUT_SYNC_ARIBA_UPLO, and click **Search**.
   
   The Proxy Class page appears.
4. From the list of proxies, click the name of the proxy, /ARBA/CO_MIOUT_SYNC_ARIBA_UPLO, that you want to configure.
   
   The Details of Consumer Proxy: <proxy name> page appears.
5. From the **Configuration** tab, click **Create Logical Port**. and choose **Manual Configuration**.
6. Under **General Configuration**, specify the following:
   - Logical Port name.
   - Description
   - Check **Logical Port is Default**.
7. Click **Next**.
   
   The New Manual Configuration of Logical Port for Consumer Proxy <proxy name> page appears.
8. From the **Authentication Settings**, choose one of the following options:
   - To use Shared Secret-based authentication, choose **User ID/Password**, and enter the User ID and the Password in the corresponding fields that appear.
   
   **Note**
   
   User ID is the name of the procurement realm and the password the shared secret that you configured in the Ariba application.
   
   - To use client certificate-based authentication, choose **X 509 SSL Client Certificate**, and specify the certificate file in the field **SSL Client PSE of transaction STRUST**. For more information, see [How to configure the client certificate for certificate-based authentication](#).
9. Click **Next**.
   
   The Transport Bindings page appears.
10. In the **Transport Bindings** page, specify the following:
   - **URL Access Path**: `/Buyer/fileupload?realm=<p2prealm>`. Realm name should be the same name that you maintain in the table `AUTH_PARAM`.
   - **Computer Name of Access URL**: Specify the path to your Ariba application. For example, `s1.ariba.com`. If you are using client certificate-based authentication, enter `certs1.ariba.com`.
   - **Port Number of Access URL**: 443
   - **URL Protocol Information**: HTTPS

11. Save the changes.

### How to configure LPCONFIG for ECC6.0 EHP3

#### Procedure

1. You must create a new RFC connection. For more information, see [How to create a new RFC connection](#) [page 158].
2. Run the transaction code `LPCONFIG`.
3. Search for the proxy class `ARBA/MIOut_Sync_AribaUpload`, The **Display/Create Logical Port** page appears.
4. Enter the port name in the field **Logical Port**.
5. Click **Create**.
6. Enter a description for the logical port.
7. Check the **Default Port**.
8. In the tab **General Settings** > **Runtime**, click **Web Service Infrastructure**.
9. In the tab **Application-Specific Settings** > **Global Settings**, check **State Management**.
10. Click the tab **Call Parameters** and choose **HTTP Destination**.
11. Specify the newly created RFC destination.
12. Choose the tab **Operations**. You can see the proxy `MIOut_Sync_AribaUpload` that has been generated.
13. In the field **SOAP Action**, enter the following URL:
14. Save and activate the entries.

### Next Steps

[How to maintain a temporary directory while running master data directly](#) [page 69].
How to create a new RFC connection

Context

Create a new RFC Connection before you configure the LPCONFIG for ECC6.0 EHP3.

Procedure

1. Run the transaction code SM59 in SAP ERP.
   
   The page Configuration of RFC Connection appears.
2. Click to choose HTTP Connection to External Server.
3. Click Create.
   
   The page RFC Destination appears.
4. Check Connection Type and choose the connection type G.
5. Enter a description in the text box Description 1.
6. Click the tab Technical Settings.
7. Enter values in the following fields:
   ○ 1. Target Host: Enter the host name of the Ariba Procurement Solution system.
8. Click the tab Logon & Security.
9. Click Basic Authentication.
10. In the section Logon, enter the following:
   ○ 1. User ID: Enter the user name for the procurement realm. For example, ERP2PTESAP_T.
   ○ 2. Password: Enter the shared secret that you specified on the Integration Toolkit Security page of your Ariba Procurement Solution.

How to create a logical file path

Context

You must create a logical file path for the ISMW_ROOT. This is required when you import master data and export transactional data for Purchase Order Header Status, Remittances, and any status updates sent for the transactions through the Direct Connectivity Integration method.
Procedure

1. Run the transaction code file in SAP ERP.
2. Click New Entries.
3. Enter ISMW_ROOT as an entry in the column Logical file path.
4. Enter a name for the logical file path ISMW_ROOT.
5. Choose the logical file path ISMW_ROOT and then double-click Logical File Name Definition, Cross-Client in the section Dialog Structure.
6. Ensure you have the following values for the logical file path ISMW_ROOT:
   - **Logical File**: ISMW_FILE
   - **Name**: ISMW_FILE-logical file name
   - **Physical file**: To be left blank without any values.
   - **Data format**: DIR
   - **Appcat.area**: To be left blank without any values.
   - **Logical path**: ISMW_ROOT
7. Save the changes.

Related Information

About integrating transactional data directly using the user interface [page 243]

How to install the SSL client certificate for HTTPS connection

Context

To enable HTTPS connection between Ariba Procurement Solution and SAP ERP, install SSL client certificate.

Procedure

1. Open your browser in Internet Explorer or Google Chrome and download the client certificate for the URL that you have configured in SOAMANAGER.
2. Click the indicator Secure Lock on the address bar and then click the tab Connection.
3. Click the link Certificate information to download the certificates.
   - The popup Certificate opens.
4. Click the tab Details and then click Copy to File.
   - The Certificate Export Wizard appears.
5. Follow the steps in the Certificate Export Wizard. Ensure that you choose the option DER encoded binary X.509 (.CER). This option is selected by default.
6. Specify the path where you want to store the certificates. Click Finish.

7. In SAP ERP, run the transaction code STRUST and click SSL Client SSL Client(Anonymous).

8. Click the Import Certificates button and then choose the path where you exported the certificates.

9. Click Allow and then Continue to download the certificates.

10. Click Add to Certificate List.

     You will see a message that you have added the certificates successfully.

     **Note**

     Ensure that you save the newly added certificates. This notifies the ICM about the new certificates you have just imported.

11. After you have saved the entries in the STRUST, run the transaction code SMICM.

12. On the menu Administration, click ICM Exit Soft and then Global.

13. In the popup Internet Communication Manager, Do you really want to restart all ICM processes in the system?, click Yes.

14. On the menu Administration, click ICM Exit Hard and then Global. Restart the services for the Exit Global.

15. In the popup Internet Communication Manager, Do you really want to restart all ICM processes in the system?, click Yes.

16. Test your connection using SOAMANAGER.

17. On the tab Search, choose Consumer Proxy from the Search By dropdown.

18. Enter /ARBA/CO* in the text box Search Pattern and choose Both Names from the Field dropdown.

19. From the list of results, choose /ARBA/CO_MIOUT_SYNC_ARIBA_UPLO.

20. On the tab Configurations, click Ping Web Service.

21. In case you see an ICM_HTTP_SSL_ERROR, run the transaction code SM59 and search for the HTTP destination. You can see the HTTP destination in the text box HTTP Destination.

22. If there is an ICM_HTTP_SSL_ERROR, run the transaction code SM59 and search for the HTTP destination.

23. Check if there is any service number available for the HTTP destination in the text box Service No.. By default, the value for this port is set to 80.

24. Delete the default service number in the text box Service No. and test the connection again.

**Results**

You see a message that the connection was successful.
How to configure the client certificate for certificate-based authentication

Procedure

1. Ensure that you have the client certificate that you have received from an Ariba trusted Certificate Authority.
2. Import the private key of the certificate into the SAP Business Site system by using the Trust Manager (transaction STRUST). You can only import certificates in the Personal Security Environment (PSE) format. Certificates in other formats must first be converted to the PSE format. Use the command line tool SAPGENPSE to do the conversion. Install the SAPGENPSE tool with the SAP Cryptographic Library installation package. For more information, see The SAP Cryptographic Library Installation Package [external document].

   For example, to convert the P12 (Public-Key Cryptography Standards) format to PSE format, enter the following command line:

   sapgenpse import_p12 -v -r <root certificate> -p <Target PSE file><Source File>.

3. Create a new Client Identity in the Trust Manager. Go to Environment > SSL Client Identities.
   - Enter “ARIBA” as the identify name and “Ariba Network Client” as the description.
   - Save the entries.
4. Import the private key of the certificate into the SAP Business Site system by using the Trust Manager (transaction STRUST).
   - Select the newly created ARIBA SSL Client ID and choose PSE > Import to import the PSE file.
   - Enter the password for the certificate, if required.
   - To save, click PSE > Save as > SSL Client.
   - Enter “ARIBA” as the SSL Client.
   - Navigate to the Own Certificate group box on the Trust Manager screen, and double click the certificate to add it to the certificate list. You can now see the newly added certificate in the Certificate List of the Trust Manager.

3. Import the root certificate into the SAP Business Suite system by using Trust Manager as follows:
   1. Double-click the SSL Client Identity ARIBA that you have created.
   2. Navigate to the Certificate group box and choose Import certificate. Click Add to Certificate List to add the imported certificate to the certificate list.
4. Use transaction SMICM to activate the changes and restart the Internet Communication Manager (ICM). Click Administration > ICM > Restart > Yes. For more information, go to the SAP documentation portal and search for the phrase “Using the ICM Monitor.”

Integrating master data using the mediated connectivity integration method

Buyers using the Ariba Procurement Solution integrated with SAP ERP can integrate master data directly from an SAP ERP system to the Ariba Procurement Solution system seamlessly using the SAP Process Integration layer through the mediated connectivity integration method.
Related Information

Integrating master data using the direct connectivity integration method from SAP ERP [page 152]

Configuring mediated connectivity

Buyers using the Ariba Procurement Solution integrated with SAP ERP can integrate master data from an SAP ERP to the Ariba Procurement Solution seamlessly using the SAP Process Integration through the mediated connectivity integration method.

The following topics list the requirement and explain the steps for setting up mediated connectivity between SAP ERP and Ariba Procurement Solution:

- About integrating master data using the mediated connectivity integration method [page 162]
- Prerequisites and limitations [page 163]
- How to configure the Integration Scenario [page 164]
- How to configure the proxy on SAP ERP systems [page 166]
- How to configure client certificate-based authentication for Mediated Connectivity [page 167]
- How to configure certificates and keys to import master data using SAP Process Integration [page 168]
- How to configure the shared secret authentication for Mediated Connectivity [page 169]
- How to configure SSL certificate for mediated connectivity [page 169]

About integrating master data using the mediated connectivity integration method

Buyers using the Ariba Procurement Solution integrated with SAP ERP can integrate master data directly from an SAP ERP to the Ariba Procurement Solution seamlessly using the SAP Process Integration through the mediated connectivity integration method.

Mediated connectivity provides enhanced security while transferring master data from an SAP ERP system to an Ariba Procurement Solution system. Ariba downloads master data from an SAP ERP to the Ariba Procurement Solution through SAP Process Integration using SOAP messages.

To integrate master data, a buyer can use either client certificates or shared secret authentication to connect to the Ariba Procurement Solution through SAP Process Integration.

If you are using the client certificate [page 167], the SAP Process Integration keystore stores the certificates and keys that you create in SAP Process Integration. You can create a view to group these certificates and keys based on views.

If you are using the shared secret-based authentication [page 169], you must configure the shared secret in the Communication Channel of SAP Process Integration.
Prerequisites and limitations

- Install the latest Ariba procurement transports and TPZ files. For more information, see How to import the Ariba Components [page 112].
- Maintain the table entries as explained in Maintaining table entries [page 105].

**Note**

If you have configured logical port for the realms to which you export master data, you must delete or deactivate the logical port configuration.

- Configure SSL client certificate for HTTPS connection. For more information, see How to configure SSL certificate for mediated connectivity [page 169].
- Specify the system ID of SAP ERP on the Service Manager > Site Manager > Customer Site > Master Data Manager > External System Configuration > Create External System window of the Ariba procurement application.
- Based on the security model you choose, configure one of the following:
  - For Shared Secret-based authentication, configure the Shared Secret settings such as the user name and password on the Service Manager > Site Manager > Customer Site > Integration Manager > Integration Toolkit Security window of the Ariba application. For more information, see How to configure the shared secret authentication for Mediated Connectivity [page 169].
  - For client certificate-based authentication, install the client certificates and configure the certificate settings at the Service Manager > Site Manager > Customer Site > Integration Manager > Integration Toolkit Security window of the Ariba application. For more information on configuring client certificate, see How to configure client certificate-based authentication for Mediated Connectivity [page 167].
- Ensure that software component ARIBA_UPLOAD 1.0 of ariba.com is available in the System, Landscape Directory (SLD).
- Set up the proxy configuration between the SAP ERP and SAP Process Integration to send and receive the proxy request and response messages. For more information, see How to configure the proxy on SAP ERP systems [page 166].

Requirements for Federated Process Control-enabled systems

1. Maintain a parent realm and a child realm and create a business system each for the parent and child realms.
2. Create a communication channel each for the parent and child realms based on the same communication channel template. Set the target URL of the communication channel for the child realm to the child realm name and the target URL of the communication channel for the parent realm to the parent realm name.
3. Create the integration scenario and specify the communication channels that you created for the parent and child realm.
4. Specify receiver determination condition for the integration scenario.

**Limitations**

Mediated Connectivity has the following limitations:

- Administrators do not get email notifications when an error occurs during the master data integration.
- Administrators can only view the log files in the runtime monitor of the SAP ERP and SAP Process Integration.
- The log files do not store a record of all the error conditions in the SLGI

**How to configure the Integration Scenario**

**Procedure**

1. From the SAP Process Integration main page, choose Integration Directory > Integration Builder and log in to the application Configuration: Integration Builder.
2. In the Integration Builder page, choose Tools > Apply Model from ES Repository.
3. From the wizard Transfer Model from ES Repository, go to Select ES Repository Model and click Process Integration Scenario.
4. Click the field Name and click Input Help from the dropdown Display Input Help.
5. Choose the following from the screen Select Process Integration Scenario from Enterprise Services Repository.
   ○ AR_Ariba_Upload
6. Click Apply. The wizard Transfer Model from ES Repository appears. The Namespace and the Software Component Version are available in the wizard.
7. Click Continue and then, click Close.
   This completes the configuration scenario.
8. In the Model Configurator, click the button Select Component View.
9. In the page Select Component View for Configuration, choose the tab Business System Components for A2A to assign components for the Ariba Procurement Solution.
10. Click the button Insert Line (+) to add a component entry in the table if no component entry line is present.
11. Click the row under the Communication Component column, then click the dropdown Value List and choose Value List <F4>. Choose the communication component name AR_Ariba_Upload from the selection list and click Apply.
12. Click the button Assign Component.
13. In the page Model Configurator, choose the tab Business System Components for A2A to assign the communication component for your Ariba Procurement Solution.
14. Click the row under the column Communication Component, then click the pulldown Value List and choose Value List <F4>. Choose the communication component name from the selection list and click Apply.
15. Click the button **Next Application Component** (the right arrow) to go to the next screen in the window.
16. Click the button **Insert Line** (‘+’) to add a component entry in the table if no component entry line is present.
17. Click the row under the column **Communication Component**, then click the pulldown **Value List** and choose **Value List ..F4**. Choose the communication component name for the business system from the selection list and click **Apply**.
18. Click the button **Save Settings** to save the changes.
19. In the page **Model Configurator**, click the button **Configure Connections**. You must assign a Communication Channel to set up the connections.
20. In the **Connections from Component Assignment** tab, click the empty **Communication Channel** cell corresponding to the to the business system component value.
21. Click the **Value List** pull down menu and choose **Value List ..F4**. Choose the communication component and click **Apply**.
22. Choose the **CC_Proxy_Sender** communication channel from the selection list and click **Apply**.

**Note**
You require the Proxy Sender communication channel only for a single stack Java machine. If you are using a dual stack Java machine, the **Communication Channel** cell must be left blank.

23. Click the **New Communication Channel** button (the icon at the top left) on the **Communication Channels** page to start the **Create Communication Channel** wizard.
24. Click **Continue** and in the next page click the **Name** field, then click the **Display Input Help** drop down list and choose **Input Help**.
   1. Choose the following Communication Channel Template:**CCT_Ariba_Upload_Receiver**
   2. Specify the communication channel name.
   3. Choose the communication component that corresponds to the business system to which you want to associate this channel.

**Note**
When you integrate master data with FPC-enabled systems, repeat these steps to create communication channels for parent and child realms. Communication channels for parent and child use the same communication channel template, **CCT_Ariba_Upload_Receiver**.

When you specify the target URL for the communication channel, ensure that you specify the URL of the child site for the channel created for the child site and the parent site URL for the one created for the parent site.

25. Click **Finish**. The Model Configurator window appears.
26. In the **Model Configurator**, click the **Create Configuration Object** button.
27. In the **Create Configuration Object** window, do the following:
   - Click **Generation** in the **General** section.
   - Choose the **Receiver Determination**, and **Interface Determination** checkboxes in the **Scope of Generation** section.
28. Click **Create New** in the **Change List for Generated Objects** section. The text box already contains an object name that you can use or modify.
29. Click **Start** to begin generating objects.
30. After the object generation is complete, click Apply to save the new configuration settings. Close the log that is created without saving. You can save it, if required.

31. On the Communication Channel page, you can specify the shared secret or client certificate authentication in the Parameters tab. Ensure that you set the parameters in the General tab - Connection Parameters section that matches your Ariba Buyer account configuration.

32. Save the changes.

Next Steps

After you create the integration scenario for integrating master data with Federated Process Control-enabled systems, edit the integration scenario to specify the following condition for receiver determination:

/p1:MT_UploadReq/Header/Parameters/Parameter/@name = realm AND /p1:MT_UploadReq/Header/Parameters/Parameter/value = <realmname>. You need to specify the condition for both parent and child realms and associate the condition with the corresponding business system communication components.

How to configure the proxy on SAP ERP systems

Procedure

1. Run the transaction code SM59 and check if following configurations exist:
   ○ SAP_PROXY_ESR under HTTP Connections to External Server
   ○ XI_INTEGRATIONSERVER_<SID> (SID - system identifier) under HTTP Connections to ABAP System (H Type RFC)
   ○ LCRAPI under TCP/IP Connections. Specify the program ID similar to the one created under Jco RFC destinations (LCRAPI_<SID>, where SID - system identifier)
   ○ SAPSLDAPI under TCP/IP Connections. Specify the program ID similar to the one created under Jco RFC destinations (SAPSLDAPI_<SID>, where SID - system identifier)

2. Run the transaction code SLDAPICUST.
3. Enter SAP XI hostname, port, XI user ID, and password.
4. Run the transaction code SXMB_ADM.
5. Navigate to Integration Engine Configuration.
6. Click Integration Engine Configuration to edit global configuration data.
7. In the Global Configuration Data section, enter the following:
   ○ Role of Business System: Application System
   ○ Corresponding Integ. Server: dest://<XI_INTEGRATIONSERVER_PPI>
     This is the same H Type RFC destination mentioned in Step 1.
8. Run the transaction code SLDCHECK.

The SAP Process Integration server appears and the Summary section indicates that the connection to XI is successful.
How to configure the proxy for single stack (Java only) instances for SAP Process Integration 7.3 and higher

Procedure

1. Run the transaction code SM59 and create RFC destination SAP_PROXY_ESR (Type G) under HTTP Connections to External Server.
   1. Under the Technical Settings tab, add AEX (PI Java Stack) server host name and port.
   2. In the Path Prefix field, enter /rep.
   3. Under the Logon & Security tab, you must ensure that the user is assigned to the role SAP_XI_REP_SERV_USER.
2. Create an HTTP destination to AEX of Type G (example: PI_AEX) as follows:
   1. Under the Technical Settings tab, add AEX (PI Java Stack) server host name and port.
   2. In the Path Prefix field, enter /XISOAPAdapter/MessageServlet?ximessage=true.
   3. Under the Logon & Security tab, user must be assigned to role SAP_XI_REP_SERV_USER.
3. Run the transaction code SXMB_ADM.
4. Navigate to Integration Engine ➤ Configuration ➤ Integration Engine Configuration ➤ Edit global configuration data.
5. Under Global Configuration Data, enter:
   ○ Role of Business System: Application System
   ○ Corresponding Integ. Server: dest://<PI_AEX>
     This is the same G Type RFC destination mentioned in step 2.
6. Click Specific Configuration and enter the following:
   ○ Category: Runtime
   ○ Parameter: IS_URL
   ○ Current Value: dest://<PI_AEX>
     This is the same G Type RFC destination mentioned in step 2.
7. Run the transaction code SLDAPICUST.
8. Enter SAP XI hostname, port, XI user ID, and password.
9. Run the transaction code SLDCHECK.

How to configure client certificate-based authentication for Mediated Connectivity

Context

For Mediated Connectivity integration that uses client certificate-based authentication, configure the certificate that you want to use for authentication.
You must download the client certificate from an Ariba-trusted Certificate Authority and store the certificate in the SAP Process Integration keystore.

Procedure

1. From the SAP Process Integration main page, go to Administration, and click Configuration Management.
2. Click Certificate and Keys.
   The page Certificate and Keys opens.
3. Click Create View and enter the view name and description.
4. Choose the created view and click the button Import Entry.
5. From the window Entry Import, choose an entry type based on the certificate and key format.
6. Based on the entry type you selected, specify the certificate attributes.
7. Click Import.
8. Click the button Edit corresponding to the communication channel and click Configure Certificate Authentication.
   The communication channel for master data integration between the Ariba Procurement Solution and SAP ERP is CCT_Ariba_Upload_Receiver.
9. In the field Target URL, specify the location of your Ariba Procurement Solution.
10. Choose the Keystore Entry and Keystore View that you have created.
11. Save and activate the communication channel.

How to configure certificates and keys to import master data using SAP Process Integration

Procedure

1. Go to SAP Process Integration Netweaver Administration and click Configuration Management.
2. Click Certificate and Keys.
3. On the page Certificate and Keys, click Create to create a new view. Create the required keys and certificates for the newly created view.
   Note
   Ensure that your required certificates are signed by an Ariba trusted Certificate Authority to successfully establish the SSL connection.
4. Go to the page SOAP Receiver Communication Channel in the tab General, check Configure Certificate Authentication.
5. In the field **Keystore Entry**, choose the key newly created in the keystore.
6. In the field **Keystore View**, choose the newly created view in the keystore.
7. In the field **Target URL**, ensure that you specify the location of the certificates stored on your SAP Process Integration.

    **Note**
    
    Ensure that your URL starts with `https://cert...

8. Save and activate the communication channel.
9. Log in to your Ariba Buyer account. Go to **Core Administrator** ➤ **Integration Tool Kit Security** ➤ **Core Administrator** ➤ **Integration Tool Kit Security** opens.
10. From the pulldown **Select the Authentication Method**, choose **Certificate**. Paste the contents of the client certificates.

**How to configure the shared secret authentication for Mediated Connectivity**

**Procedure**

1. Open the Configuration Integration Builder from the main page of SAP Process Integration.
2. In the **Objects** tab of the Integration Builder, click **Communication Channel**.
3. Click **Edit** button that corresponds to the **Communication Channel**, and click **Configure User Authentication**.
4. In the **Target URL** field, specify the URL of the procurement system. For Example: `http://<URL to the Ariba procurement solution>/Buyer/fileupload?<realm><realm name that you maintain in `/ARBA/AUTH_PARAM`>` such as, `https://buyer.ariba.com//Buyer/fileupload?<realm2p>`
5. In the **User** text box, enter the realm name for your procurement system.
6. In the **Password** text box, enter the same shared secret that you specified on the **Integration Toolkit Security** page of your Ariba procurement system.
7. Save and activate the communication channel.

**How to configure SSL certificate for mediated connectivity**

**Context**

To enable HTTPS connection between SAP Process Integration and Ariba Procurement Solution over Mediated Connectivity, configure SSL certificate.
Procedure

1. To download the SSL client certificate for the URL Ariba Procurement Solution, follow these steps.
   1. Open your browser in Internet Explorer or Google Chrome, and download the SSL client certificate for the URL that you configured in the SOAP communication channel.
   2. Click the indicator Secure Lock on the address bar and then, click the tab Connection.
   3. Click the link Certificate information to download the certificates. The popup Certificate opens.
   4. Click the tab Details and then click Copy to File. The Certificate Export Wizard opens.
   5. Follow the steps in the Certificate Export Wizard. Ensure that you choose the option DER encoded binary X.509 (.CER). This option is selected by default.
   6. Specify the path where you want to store the certificates and click Finish.
2. The SSL client certificates that SOAP Adapter uses to connect to Ariba Procurement Solution must be stored in the SAP Process Integration keystore.
   1. Go to the SAP Process Integration main page and click Administration > Configuration Management.
   2. Click Certificate and Keys.
   3. Search for the default keystore TrustedCAs and click the button Import Entry.

   **Note**

   Alternatively, you can add the SSL client certificate to the keystore you created for client certificate-based authentication.

   6. Choose the option X.509 Certificate from the dropdown Select Entry Type in the window Entry Import.
   7. Choose file and import.
Integrating transactional data

Integrating transactional data using the file channel [page 171]
Integrating transactional data using the web services channel [page 200]
Integrating transactional data directly using the user interface [page 243]
Integrating transactional data using the mediated connectivity integration method [page 250]

Integrating transactional data using the file channel

Transaction data includes information like remittance advice, invoice, purchase orders, and receipts. You import/export data using scheduled integration events, or run integrations events manually from Ariba Administrator.

- Transactional data integration event process flow [page 171]
- Advanced pricing details [page 177]
- Exporting purchase orders [page 178]
- Exporting change purchase orders [page 180]
- Exporting close purchase orders [page 183]
- Exporting cancel purchase orders [page 185]
- Importing purchase order header status [page 186]
- Exporting receipts [page 187]
- Exporting invoices [page 188]
- Exporting expense reports [page 193]
- Accommodating custom fields [page 194]
- Calling the SAP BAPI [page 195]
- Scheduling [page 196]
- Importing remittance advice [page 196]

Transactional data integration event process flow

An executable program is defined for each file channel integration event. This program can be scheduled to run in the background listening to input CSV data files from the Ariba Procurement Solution and export it to SAP ERP database.
The following illustration shows a typical data integration process flow using the file channel:

Transactional data is downloaded into CSV file formats by executing the appropriate transaction integration events. The Data Transfer Tool then picks up these CSV files and moves them into the SAP ERP database.

The custom RFC programs are provided to read the data from CSV files on the SAP database. Invoke the SAP RFC/BAPI to create the data in SAP ERP, and send the response back to the Ariba Procurement Solution.

**Figure 6: Typical Data Integration Process Flow Using the File Channel**

**Configuration tasks**

- Importing SAP transports [page 172]
- File channel integration - program structure [page 173]
- Working with logical file names [page 175]

**Importing SAP transports**

Integrating transactional data using the file channel involves importing of SAP transports as a prerequisite.

These SAP transports contain SAP Programs, RFCs, and supporting structures. The SAP executable programs are used to move the exported data into SAP ERP. Each program internally calls the BAPI function. The BAPI function facilitates the moving of data into SAP ERP.
For more information on how to import SAP transports, see the *Installation Guide for Ariba Procurement Solution Integrated with SAP ERP*.

**File channel integration - program structure**

The following diagram illustrates how the SAP programs are wrapped around the BAPI function modules.

![Diagram](image-url)

Each stage (layer) in this illustration is acted upon by an executable program and is explained briefly below.

*Figure 7: How SAP Programs Wrap Around BAPI Function Modules*
Selection Screen

This program comprises five parameters. These parameters are briefly described in the following table:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logical File Name</td>
<td>Defines the logical path and the physical location of the CSV files.</td>
</tr>
<tr>
<td>Directory Separator</td>
<td>Physical separator for directories in SAP ERP.</td>
</tr>
<tr>
<td>Encoding in response files</td>
<td>Encoding technique that is used. This is UTF-8 by default.</td>
</tr>
<tr>
<td>Variant</td>
<td>Variant Name.</td>
</tr>
<tr>
<td>Partition</td>
<td>Partition Name.</td>
</tr>
</tbody>
</table>

Read CSV Files

At this stage, the executable program reads the CSV files that are specified in the logical path directory and inserts them into the internal tables.

Populate Data from Internal Tables to RFC Structures

Here, program picks up the data from internal tables and moves it to the appropriate RFC Function Module.

Call RFC Function Module

This program exports the data into the SAP ERP database.

RFC Returns Response Messages

In this final stage, the messages are written into CSV files. If the export is successful, the messages are written to the NumberImport CSV files. If there are errors in the export event, messages are written to the ErrorImport CSV files and placed under the Response directory.

Move CSV Files to Processed Folder

The corresponding program finally moves the incoming CSV files into a processed folder. The output is displayed with corresponding Order Numbers.
Working with logical file names

A logical file name defines the logical path and the physical location of the CSV files in SAP ERP. Every transaction has a default Logical File Name.

You can choose to modify the logical path to the file directory, or create a new logical file name to overwrite the default logical file name. To do this, execute the corresponding program (for transactional data integration) and specify the new logical file name that you are creating, and save it as a variant.

Note
The physical path to the logical file name should not exceed more than 50 characters.

Ensure the CSV file is placed under the incoming ERPOrder directory, which is preceded by the logical file path. For example, if the logical file path is

C:\usr\sap\<SYSID>\DVEBMGS00\work\POPush\<CSV File Name>

you must create an incoming directory in the server and place the CSV file in this directory, as shown:

C:\usr\sap\<SYSID>\DVEBMGS00\work\POPush\incoming\ERPOrder\<FileName>

Here:
- \incoming: is common to all transactions
- ERPOrder: is specific to each transaction.

Note
- Do not change the ERPOrder directory name. You can choose to extend this directory name, for example, ERPOrder_FOO.
- The number of characters comprising the full path (for example, C:\usr\sap\<SYSID>\DVEBMGS00\work\POPush) must not exceed 50 characters.

During run time, the following two directories are created under the Root directory:

- processed
- response

For example:

C:\usr\sap\<SYSID>\DVEBMGS00\work\POPush\processed
C:\usr\sap\<SYSID>\DVEBMGS00\work\POPush\response

The processed directory contains the executed CSV files (that are moved from the incoming directory location to here), and the response directory contains the response CSV files for the Ariba Procurement Solution.

Configuring the options file

When running any export integration event, the corresponding SAP program picks up the CSV files from a predetermined location. This directory path must be configured in the Options file of the Data Transfer Tool.
Configure the Options file as follows:

- Include `\incoming` at the end of the `downloadDir` option.
- Specify the `downloadDir` and `filePrefix` options.

### Scheduling Data Transfer Tool

In a scenario where there are very frequent transactions occurring, you can schedule the Data Transfer Tool to run at regular time intervals to automatically download the transaction CSV files and export them to the SAP database.

You can use any third-party scheduler, such as the one that is provided with your operating system. You must also schedule the corresponding SAP program to automatically pick up the CSV files every time the files are downloaded by the Data Transfer Tool.

### Transaction data integration event components

The following table lists the various components that constitute every transactional data integration event.

<table>
<thead>
<tr>
<th>Transactional Data</th>
<th>Event Name</th>
<th>SAP Program</th>
<th>RFC Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Purchase Order</td>
<td>PurchaseOrder</td>
<td>/ARBA/CR_BAPI_PO_CREATE</td>
<td>/ARBA/BAPI_PO_CREATE</td>
</tr>
<tr>
<td></td>
<td>Export</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Changed Purchase</td>
<td>PurchaseOrder</td>
<td>/ARBA/CR_BAPI_PO_CHANGE</td>
<td>/ARBA/BAPI_PO_CHANGE</td>
</tr>
<tr>
<td>Order</td>
<td>ChangeExport</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Canceled Purchase</td>
<td>PurchaseOrder</td>
<td>/ARBA/CR_BAPI_PO_CANCEL</td>
<td>/ARBA/BAPI_PO_CANCEL</td>
</tr>
<tr>
<td>Order</td>
<td>CancelExport</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Purchase Order</td>
<td>PurchaseOrder</td>
<td>/ARBA/CR_BAPI_PO_CANCEL</td>
<td>/ARBA/BAPI_PO_CANCEL</td>
</tr>
<tr>
<td>Header Status</td>
<td>HeaderStatusImport</td>
<td>/ARBA/CR_PO_HEADER_STATUS</td>
<td>/ARBA/PO_HEADER_STATUS</td>
</tr>
<tr>
<td>Receipts</td>
<td>ReceiptExport</td>
<td>/ARBA/CR_GR_CREATE</td>
<td>/ARBA/GOODS_RECEIPTS_CREATE</td>
</tr>
<tr>
<td>Invoice</td>
<td>PaymentExport</td>
<td>/ARBA/CR_INVOICE_CREATE</td>
<td>/ARBA/BAPI_INVOICE_CREATE</td>
</tr>
<tr>
<td>Remittance</td>
<td>RemittanceImport</td>
<td>/ARBA/CR_REMITTANCE_EXPORT</td>
<td>/ARBA/REMITTANCE_EXPORT</td>
</tr>
<tr>
<td>Requisition Submit</td>
<td>RequisitionSubmit</td>
<td>/ARBA/CR_BAPI_PO_CREATE</td>
<td>/ARBA/BAPI_PO_CREATE</td>
</tr>
<tr>
<td></td>
<td>RealTimeBudgetExport</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Requisition Revert</td>
<td>RequisitionRevert</td>
<td>/ARBA/CR_REMITTANCE_EXPORT</td>
<td>/ARBA/REMITTANCE_EXPORT</td>
</tr>
<tr>
<td></td>
<td>RealTimeRevertBudgetExport</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transactional Data</td>
<td>Event Name</td>
<td>SAP Program</td>
<td>RFC Used</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------</td>
<td>----------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>Budget Check</td>
<td>RequisitionBudgetCheckExport</td>
<td></td>
<td>/ARBA/ACC_PURCHASE_REQUI_CHECK</td>
</tr>
<tr>
<td></td>
<td>• Requisition Derive</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Invoice Derive</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• RequisitionDeriveAccountingExport</td>
<td></td>
<td>/ARBA/DERIVATION</td>
</tr>
<tr>
<td></td>
<td>• InvoiceReconciliationDeriveAccou</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ntAccountingExport</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Create Advance Pay-</td>
<td>ExportAdvancePayments</td>
<td>/ARBA/CR_ADV_PAYMENT_CREATE</td>
<td>/ARBA/ADV_PAYMENT_POST</td>
</tr>
<tr>
<td>ment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cancel Advance Pay-</td>
<td>ExportCancelAdvancePayments</td>
<td>/ARBA/CR_ADV_PAYMENT_CANCEL</td>
<td>/ARBA/ADV_PAYMENT_REV_POST</td>
</tr>
<tr>
<td>ment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Advance Payment Rem-</td>
<td>ImportAdvancePaymentRemittance</td>
<td>/ARBA/CR_ADV_PAY_REMIT_EXPORT</td>
<td>/ARBA/ADV_PAY_REMIT_EXPORT</td>
</tr>
<tr>
<td>ittance</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**i Note**
The Ariba Procurement Solution integrated with SAP ERP does not support purchase orders or invoices containing service line items.

### Advanced pricing details

While creating a requisition in the Ariba Procurement Solution, the user can specify advanced pricing details for a line item or multiple line items. Advanced pricing details includes:

- Quantity Based Pricing
- Unit Conversion Pricing

Quantity based pricing allows the unit price of an item to be based on a different price unit quantity than 1. Unit Conversion Pricing allows unit of measure conversion in the pricing calculation, when the unit of measure on the order differs from the pricing unit of measure.
Exporting purchase orders

Purchase orders that are in the ordering state are exported to SAP ERP. The Ariba Procurement Solution writes the exported purchase order details into five CSV files:

- PurchOrdAcctDetails.csv
- PurchOrdHeaderDetails.csv
- PurchOrdLineDetails.csv
- PurchOrdScheduleDetails.csv
- PurchOrdLineAddDetails.csv

While creating a requisition in the Ariba Procurement Solution, if the user has specified advanced pricing details for a line item, then in the exported PurchOrdLineDetails.csv file, the following fields are added:

- PriceBasisQuantity
- PriceBasisQuantityUOM
- ConversionFactor
- PriceBasisQuantityDesc

Advanced pricing details are also applicable for change purchase order and payment requests. Therefore, when a change PO and payment request are exported, the above mentioned fields are available in the ChangePurchOrdLineDet.csv and PaymentLineItemDet.csv files respectively.

A sample CSV file with the above mentioned fields appears as follows:

```
"UTF-8"

PurchOrdLineDetails.csv

"ERPOrderID","RequisitionRequester","Due_Time","BillTo","ServiceLineItemDetails","ShortName","SupplierPartAuxiliaryID","ShipTo","ERPCommodityCode","ConversionFactor","UnitPrice","Line_Amount","Quantity","PriceBasisQuantityUOM","Description","SupplierLocation","ReceivingType","ReqID","Currency_cd","SupplierPartNumber","PriceBasisQuantityDesc","UOM","SupplierLocationName","PriceBasisQuantity","SupplierName","CommodityCode","DeliverToOrigSysRefID","OrigSysLineNumber","Line_Amount_InBaseCurrency","POLineNum"
"ERP1","ADeep","4:00:00 PM","2445","PO1292464699548","1","1","AdHocItem","PR313","adeep","AdHocItem","","RANDOM123","EA","432111503","150.0000",""USD","150.0000","100","15000.0000",""USD","15000.0000","US002","US017","ArnoldDeep","","","",1,"",
```

To ensure that the above mentioned fields related to advanced pricing details are processed in SAP ERP, download the relevant SAP ERP transport requests from https://connect.ariba.com and import them into your SAP ERP.

To perform a purchase order export event, you run the Ariba Data Transfer Tool from the SAP server. The Data Transfer Tool downloads these CSV files from Ariba Procurement Solution and places them in a location that is accessed by the SAP program /ARBA/CR_BAPI_PO_CREATE. This SAP program picks up these CSV files and transfers them into the SAP database.

Related Information

Integrating taxes and charges [page 227]
How to run the purchase order export event

Prerequisites

The purchase order integration with Ariba Procurement Solution does not provide support for the following features:

- Exporting tax code information
- Split accounting for service-line items
- Ariba procurement solutions export field mapping mechanism.

If you need to add Ariba Procurement Solution extrinsic fields to RFC objects, you add these fields to the RFC structures. Next, you add this field in the AML mapping, re-import the schema and make appropriate changes in the mapping.

If a purchase order or a change purchase order is exported using a RFC, the carriage returns in the line item text are exported and displayed as the hash character ('#') in the line item text in the SAP ERP purchase order.

- You cannot export amount-based purchase orders with line items that have a zero price.

Procedure

1. Create a purchase order (PO) with multiple line items in Ariba Procurement Solution.

   **Note**
   When creating the PO, the NeedByDate field is not mandatory. However, when you push the PO to SAP ERP, this field is required else the PO push fails.

2. In the SAP ERP, use the SAP Transaction code SE38 to start the SAP user interface and run the `/ARBA/CR_BAPI_PO_CREATE` program.

3. Specify the parent directory path and folder to which the Data Transfer Tool downloads the CSV files from Ariba Procurement Solution. The `/ARBA/CR_BAPI_PO_CREATE` program also requires a logical file path as an input parameter. This logical file path must be the same as the one specified in the Data Transfer Tool and is the location where the CSV files are located.

   **Note**
   ○ You can create a new logical file name (use ZARIBA_POPUSH_LOG_FNAME, which is the default value) using the FILE transaction. When running this transaction, you must specify a logical path and a physical file name. See Working with logical file names [page 175].
   The parent directory that you specify must contain a sub-directory called incoming. The Data Transfer Tool loads the CSV files into the incoming directory.
   ○ This directory name and path must match the `%downloadDir%\%filePrefix%` directory that you specified in the Data Transfer Tool Options file.

4. Specify the path directory separator value (\ or /).

5. To schedule the `/ARBA/CR_BAPI_PO_CREATE` program to run at regular (or pre-defined) time intervals, save the program input as a variant, which you can later specify to be run as a job.
You can schedule this program using the SAP ERP SM36 transaction.

6. When this program is executed, it returns the response records in CSV file format and places these files in the /response directory.

Results

- If the export event is successful, the program returns a NumberImportxxxxxx.csv file (where xxxxxx is the timestamp).
- If the export event failed due to errors, the program returns an ErrorImportxxxxxx.csv file (where xxxxxx is the timestamp).

Adjusting the Need By Date

When exporting purchase orders, the Need By Date field value contained only the date and not the time or time zone information. Because of time zone differences, the Need By Date specified while creating a requisition was not getting stored correctly in the Ariba server. As a result, when you executed the Export Purchase Order to ERP integration task, incorrect date was getting exported. This behavior was seen in both file-based and web services-based integrations.

To ensure that the correct date is stored in Ariba Server, you can adjust the Need By Date based on the time zone of the requester. To achieve this, new fields called DLV_DT and SAPNeedByDate have been added to file-based and web services-based integrations respectively.

Note

You need to apply this fix only if you want the Need By Date value adjusted to the time zone of the requester.

To adjust the Need By Date based on the time zone of the requester, you must:

1. Import the relevant SAP Transports.
2. Import the SAP Process Integration design package for purchase orders into your SAP Process Integration system.

You can download the changed SAP transports and the design package for purchase orders from https://connect.ariba.com.

Exporting change purchase orders

Purchase orders that are in the ordering state are exported from Ariba Procurement Solution to SAP ERP. After the export, the exported change purchase order details are available in the following seven CSV files:

- ChangePurchOrdAcctDet.csv
- ChangePurchOrdDelAcct.csv
- ChangePurchOrdDelItems.csv
To perform a change/cancel purchase order export event, you run the Ariba Data Transfer Tool from the SAP server. The Data Transfer Tool downloads these CSV files from Ariba Procurement Solution and places them in a location that is accessed by the /ARBA/CR_BAPI_PO_CHANGE SAP program. This SAP program picks up these CSV files and transfers them into the SAP database.

The following are important notes on exporting changed and canceled purchase orders to SAP ERP:

- If you enable the change/cancel purchase order feature, users must always change and cancel orders through Ariba Procurement Solution and not change or cancel them through SAP ERP. Therefore, apply the appropriate authorization concept in SAP ERP to prevent users from changing purchase orders directly in SAP ERP.
- To enable the change/cancel purchase order feature, you install the change/cancel order transport. The transport number is listed in the transport readme.txt file. For more information see the Installation Guide for Ariba Procurement Solution Integrated with SAP ERP.
- A limitation in the standard SAP BAPI creates the following situation: If the Account Assignment category is modified in Ariba Procurement Solution, the Account Assignment category is changed in SAP ERP. Furthermore, the GLAccount is changed to a default GLAccount after it has been pushed to SAP ERP. To resolve this issue, modify the GLAccount in SAP ERP using the Account Assignment tab in the Item Detail section.
- On SAP ERP service orders, you cannot change the line items, but you can delete them or cancel the entire order.
- To debug the change/cancel purchase order feature, set the parameter /ARBA/DEBUG_CPO to X and set Z_ARIBA_CPO_FNAME to the name of a log file where the change/cancel order information is logged. You set both of these parameters in the table /ARBA/TVARV.
- Plant data is not editable if the purchase order line Item is partially or fully received or partially or fully invoiced in Ariba Procurement Solution or SAP ERP.
- Changing the net price of a zero amount line item in a purchase order to a non-zero value is not supported.
- You cannot export amount-based purchase orders with line items that have a zero price.

Related Information

Integrating taxes and charges [page 227]

How to run the change purchase order export event

Procedure

1. In SAP ERP, use the SAP Transaction code SE38 to start the SAP user interface and run the /ARBA/CR_BAPI_PO_CHANGE program.
2. Specify the parent directory path and folder to which the Data Transfer Tool downloads the CSV files from Ariba Procurement Solution. The /ARBA/CR_BAPI_PO_CHANGE program also requires a logical file path as an input parameter. This logical file path must be the same as the one specified in the Data Transfer Tool and is the location where the CSV files are located.

**Note**
- You can create a new logical file name (use ZARIBA_CHANGEPO_LOG_FILENAME, which is the default value) using the FILE transaction. When running this transaction, you must specify a logical path and a physical file name. For more information, see [Working with logical file names](page 175).
- The parent directory that you specify must contain a sub-directory called incoming. The Data Transfer Tool loads the CSV files into the incoming directory.
- This directory name and path must match the %downloadDir\%\filePrefix% directory that you specified in the Data Transfer Tool Options file.

3. Specify the path directory separator value (\ or /).

4. To schedule the /ARBA/CR_BAPI_PO_CHANGE program to run at regular (or pre-defined) time intervals, save the program input as a variant, which you can later specify to be run as a job.

You can schedule this program using the SAP SM36 transaction.

5. When this program is executed, it returns the response records in CSV file format and places these files in the /response directory.

**Results**

- If the export event is successful, the program returns a NumberImportxxxxx.x.csv file (where xxxxxx is the timestamp).
- If the export event failed due to errors, the program returns an ErrorImportxxxxx.csv file (where xxxxxx is the timestamp).

**Payment terms integration**

The payment terms information is now exported as part of the purchase order (PO) and change purchase order (CPO) transactions and sent to the SAP ERP system. To export payment terms, a new field called PaymentTerms is added to the PurchOrdHeaderDetails.csv and ChangePurchOrdHeaderDet.csv files.

When a user creates a requisition in the Ariba Procurement Solution, the payment terms information associated with the specified supplier is selected. When the purchase order or change purchase order is exported, this value is populated to the new field PaymentTerms in the respective CSV file.
Exporting close purchase orders

Ariba Cloud Integration Release 9.0 enables buyers using Ariba Procurement Solution integrated with SAP ERP to integrate purchase order closure information from Ariba Procurement Solution with SAP ERP.

Integration of purchase order closure information between Ariba Procurement Solution and SAP ERP ensures that the purchase order information in Ariba Procurement Solution and SAP ERP is in sync, and improves the fund allocation and utilization. When a purchase order in SAP ERP is marked as closed, SAP ERP releases any unused fund that was allocated for the purchase order and sets any open quantity to zero.

However, note that the integration also updates status of corresponding purchase orders in SAP ERP when buyers reopen purchase orders in Ariba Procurement Solution. If a buyer reopens a closed purchase order in Ariba Procurement Solution, SAP ERP returns any fund that was released as unused when the purchase order was closed.

You can schedule the program /ARBA/CR_BAPI_PO_CLOSE in SAP ERP to export the purchase order closure information from Ariba Procurement Solution to SAP ERP. Alternatively, you can run the transaction code /ARBA/PO_CLOSE

The program exports the purchase order closure information from Ariba Procurement Solution as a CSV file, updates the status of corresponding purchase orders in SAP ERP, and returns status messages to Ariba Procurement Solution.

In Ariba Procurement Solution, buyers can specify a status while they close a purchase order. The following table lists the status messages that buyers can set in Ariba Procurement Solution and the corresponding status update in SAP ERP:

<table>
<thead>
<tr>
<th>Status in Ariba Procurement Solution</th>
<th>Status in SAP ERP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order Closed for Receiving</td>
<td>Delivery Complete</td>
</tr>
<tr>
<td>Order Closed for Invoicing</td>
<td>Final Invoice</td>
</tr>
<tr>
<td>Order Closed for All Actions</td>
<td>Delivery Complete and Final Invoice</td>
</tr>
</tbody>
</table>

**Note**

SAP ERP does not support the Order Closed and Adjusted for Receiving and Closed for Change status for purchase orders. Purchase orders closed with either of these status messages in Ariba Procurement Solution are not updated in SAP ERP.

When buyers reopen purchase orders that were previously-closed in Ariba Procurement Solution, the status check boxes, such as Delivery Complete or Final Invoice or both, in SAP ERP for the corresponding purchase orders are also cleared.

For the CSV files that the program /ARBA/CR_BAPI_PO_CLOSE generates, you must configure the logical file name 2ARIBA_POCLOSE_LOG_FNAME, and set its physical path to <local drive, for example D:\usr\sap\<SYSID>\DVEBMGS00\work\POCLOSE\, where SYSID is the ERP system ID.

You also need to create the following directories at the physical location you specified for the logical file name:

- **Incoming**, for the /ARBA/CR_BAPI_PO_CLOSE program to post the incoming CSV file that contains information about purchase orders that are closed. Within the Incoming directory, the /ARBA/
CR_BAPI_PO_CLOSE program creates a directory with the current timestamp in the YYYYMMDDHHMMSS format as its name, and saves the CSV file, ClosePurchOrdLineDetails.csv, to the newly-created directory.

**Processed**, for the /ARBA/CR_BAPI_PO_CLOSE program to move the system-created directory and the ClosePurchOrdLineDetails.csv file after the program updates the purchase orders in SAP ERP.

**Response**, for the /ARBA/CR_BAPI_PO_CLOSE program to save the status updates that the program sends to Ariba Procurement Solution as CSV files. For successful updates, the /ARBA/CR_BAPI_PO_CLOSE creates the following CSV files:

- PurchaseOrderCloseStatusHeaderImport.csv
- PurchaseOrderCloseStatusItemImport.csv

In case of an error, the /ARBA/CR_BAPI_PO_CLOSE program generates one CSV file, PurchaseOrderCloseStatusErrorImport.csv.

The files are saved to a directory that /ARBA/CR_BAPI_PO_CLOSE program creates within the Response directory. The newly-created directory uses the timestamp as its name.

After the /ARBA/CR_BAPI_PO_CLOSE program updates the purchase orders in SAP ERP, the program copies the system-created directory and the CSV file received from Ariba Procurement Solution to the Processed directory and deletes the same from the Incoming directory. However, the contents of the Processed and Response directories are retained until the directories and files are manually deleted.

### Prerequisites

- Configure the purchase order integration and change purchase order integration between Ariba Procurement Solution and SAP ERP.
- Enable the following parameters in Ariba Procurement Solution:
  - Application.Purchasing.PullCloseOrderStatusAckFromERP
  - Application.Purchasing.PushCloseOrderStatusToERP
  - Configure the logical file name ZARIBA_POCLOSE_LOG_FNAME, and set its physical path to <local drive>, for example D:\usr\sap\<SYSID>\DVEBMGS00\work\POCLOSE\, where SYSID is the ERP system ID.
  - Create the following directories at the physical location you specified for the logical file name:
    - Incoming
    - Processed
    - Response

### Limitations

- SAP ERP does not support the Order Closed and Adjusted for Receiving and Closed for Change status for purchase orders. Purchase orders closed with either of these status flags in Ariba Procurement Solution are not updated in SAP ERP.
- SAP ERP does not support the Delivery Complete status for service purchase orders. Because of this limitation, SAP ERP does not update service purchase orders closed in Ariba Procurement Solution with the status Order Closed for Receiving. Similarly, for service purchase orders that are closed in Ariba
Procurement Solution with the status Order Closed for All Actions, only the Final Invoice check box is checked in SAP ERP.

Exporting cancel purchase orders

Purchase orders that are in the ordering state are exported from the Ariba Procurement Solution to SAP ERP. After the export, the exported cancel purchase order details are available in the CancelPurchOrdHeader.csv file.

To perform a cancel purchase order export event, you run the Ariba Data Transfer Tool from the SAP server. The Data Transfer Tool downloads these CSV files from the Ariba Procurement Solution and places them in a location that is accessed by the SAP program /ARBA/CR_BAPI_PO_CANCEL. This SAP program picks up these CSV files and transfers them into the SAP database.

How to run the cancel purchase order export event

Procedure

1. In SAP ERP, use the SAP Transaction code SE38 to start the SAP user interface and run the /ARBA/CR_BAPI_PO_CANCEL program.

2. Specify the parent directory path and folder to which the Data Transfer Tool downloads the CSV files from Ariba Procurement Solution. The /ARBA/CR_BAPI_PO_CANCEL program also requires a logical file path as an input parameter. This logical file path must be the same as the one specified in the Data Transfer Tool and is the location where the CSV files are located.

   Note
   ○ You can create a new logical file name (use ZARIBA_CANCELPO_LOG_FNAME, which is the default value) using the FILE transaction. When running this transaction, you must specify a logical path and a physical file name. For more information, see Working with logical file names [page 175]. The parent directory that you specify must contain a sub-directory called incoming. The Data Transfer Tool loads the CSV files into the incoming directory.
   ○ This directory name and path must match the %downloadDir%\%filePrefix% directory that you specified in the Data Transfer Tool Options file.

3. Specify the path directory separator value (\ or /).
4. To schedule the /ARBA/CR_BAPI_PO_CANCEL program to run at regular (or pre-defined) time intervals, save the program input as a variant, which you can later specify to be run as a job.

   You can schedule this program using the SAP SM36 transaction.
5. When this program is executed, it returns the response records in CSV file format and places these files in the /response directory.
Results

- If the export event is successful, the program returns a `NumberImportxxxxxx.csv` file (where `xxxxxx` is the timestamp).
- If the export event failed due to errors, the program returns a `CancelPOErrorpullXXXXXX.csv` file (where `xxxxxx` is the timestamp).

Importing purchase order header status

When changed orders are in the `Received`, `Receiving`, `Invoiced`, or `Invoicing` state, the Ariba Procurement Solution cannot cancel the orders. Similarly, when purchase orders have been received or invoiced in SAP ERP, the cancel export event to SAP ERP fails.

If an order reaches SAP ERP successfully, its status can still change in SAP ERP. For example, if you receive against an order in SAP ERP, a subsequent cancel order from the Ariba Procurement Solution is certain to be rejected. To reduce the occurrence of such situations, the scheduled task `ProcessERPHeaderStatusPull` imports header status information from SAP ERP. This task sets the `ERPAllowCancel` field, which specifies whether or not the order can be canceled.

**Note**

- Use `ProcessERPHeaderStatusPull` only if you have enabled change and cancel order exports to SAP ERP, and you are doing a lot of receiving in SAP ERP.
- The `ProcessERPHeaderStatusPull` scheduled task does not eliminate all the cancel export failure errors. SAP ERP uses many factors to determine if it is possible to cancel an order. However, the status import is based on the invoicing and receiving status. This is applicable for the web services channel. For file channel, this is a batch data import and pulled as part of status import.

`ProcessERPHeaderStatusPull` imports the status for all orders that are in the `Ordered` state for the time period (days before the current date) you specify in the `QueryPeriod` parameter.

Running the purchase order header status import

The `/ARBA/CR_PO_HEADER_STATUS` RFC program is used to import the status information from SAP ERP. It downloads the `PurchaseOrderERPHeaderStatusFileImport.csv` file. The file path would be based on the logical file name defined on the selection screen. This file is picked up when the `ProcessERPHeaderStatusPull` schedule task is run in the Ariba Procurement Solution.

You can also specify the schedule for the `ProcessERPHeaderStatusPull` task in the `ScheduledTask.table`. 
Exporting receipts

The files received in the Ariba Procurement Solution are:

- ReceiptHeaderDetails.csv
- ReceiptLineDetails.csv

To perform a receipt export event, you run the Ariba Data Transfer Tool from the SAP server. The Data Transfer Tool downloads these CSV files from the Ariba Procurement Solution and places them in a location that is accessed by the SAP program /ARBA/CR_GR_CREATE. This SAP program picks up these CSV files and transfers them into the SAP ERP database.

How to run the receipts export event

Procedure

1. Run the transaction code SE38 in SAP ERP to start the SAP user interface and run the program /ARBA/CR_GR_CREATE.
2. Specify the parent directory path and folder to which the Data Transfer Tool downloads the CSV files from Ariba Procurement Solution. The program /ARBA/CR_GR_CREATE also requires a logical file path as an input parameter. This logical file path must be the same as the one specified in the Data Transfer Tool and is the location where the CSV files are located.
   
   **Note**
   - You can create a new logical file name (use ZARIBA_GRPUSH_LOG_FNAME, which is the default value) by running the transaction code FILE. When running this transaction, you must specify a logical path and a physical file name. See Working with logical file names [page 175].
   - The parent directory that you specify must contain a sub-directory called incoming. The Data Transfer Tool loads the CSV files into the directory incoming.
   - This directory name and path must match the directory %downloadDir%\%filePrefix% that you specified in the Data Transfer Tool file Options.
3. Specify the path directory separator value (\ or /).
4. To schedule the program /ARBA/CR_GR_CREATE to run at regular (or pre-defined) time intervals, save the program input as a variant, which you can later specify to be run as a job.
   
   You can schedule this program by running the transaction coder SM36.
5. When this program is executed, it returns the response records in CSV file format and places these files in the directory /response.

Results

- If the export event is successful, the program returns a NumberImportxxxxxx.csv file (where xxxxxxx is the timestamp).
• If the export event failed due to errors, the program returns an `ErrorImportxxxxxx.csv` file (where `xxxxxx` is the timestamp).

**Filtering and BAdI methods**

Customized fields must be extended in the `zxtgrheadr` and `/ARBA/GOODS_RECEIPTS_ITEM` custom structures that are provided in the program.

**Custom error process routines**

The `customer_exit_gr_error` routine is available in the `/ARBA/CR_GR_CREATE` program. If you want to perform any other error data processing on the internal tables, you must do so before writing them on to the CSV file.

**Limitations for exporting receipts**

In a receipt, if you have both accepted quantity and rejected quantity, only accepted quantity is integrated with the SAP ERP system. All rejected quantities are ignored.

**Exporting invoices**

After an invoice has been approved, the final step is to send a payment to the supplier. When it creates an invoice reconciliation document, the Ariba Procurement Solution also creates a new approvable document, called a payment request, to track the payment process.

The Ariba Procurement Solution exports payment requests that use the External System payment model only. The Ariba Procurement Solution does not export partial payments, even if you import a partial remittance from your SAP ERP.

**Related Information**

[Integrating non-PO (FI) invoices](#) [page 232]
[Integrating taxes and charges](#) [page 227]
Prerequisites for exporting payments

Ensure the following prerequisites are met before proceeding with the payment export:

1. Tax Data is populated (includes TaxCodePull, TaxCodeLookup, TaxRateLookupBuTaxCode)
2. Remittance Location Information is loaded
3. Payment Term and Method are loaded

In general, the process works as follows:

- The Ariba Procurement Solution writes the reconciled invoices to the following CSV files:
  - PaymentAccountDet.csv
  - EnhancedPaymentAccountDet.csv
  - PaymentGLAccDet.csv
  - PaymentHeaderExport.csv
  - PaymentLineItemDet.csv
  - PaymentTaxExport.csv

For information about these files, see Understanding SAP Payment Data in the Ariba Procurement Data Import and Administration Guide. You configure the directory path to these files in the Ariba Data Transfer Tool.

- On the SAP server, run the Ariba Data Transfer Tool, which requests the Ariba Procurement Solution to download the CSV files.
- Run the SAP ERP program /ARBA/CR_INVOICE_CREATE, which transfers the invoice data from the CSV files into the SAP ERP database. The response files are stored in the directory Response.

Related Information

Prerequisites [page 231]

How to run the payments export event

Context

The input for /ARBA/CR_INVOICE_CREATE includes the directory where the CSV files should be written and the path separator character. The file location is specified in the Ariba Data Transfer Tool options file for this download. Set the options file as follows:

- Use the Ariba Data Transfer Tool Event ExportPaymentFile.
- The downloadDir option must include \incoming at the end.
- The downloadDir + filePrefix options.
Procedure

1. In SAP ERP, run the program /ARBA/CR_INVOICE_CREATE using SAP transaction code SE38 to start the SAP user interface for this program.

2. Specify the parent directory path and folder to which the Ariba Data Transfer Tool writes the invoice CSV files. ZARIBA_INVOICE_LOG_FNAME expects as input, a logical file path, which indicates where the Ariba Data Transfer Tool has downloaded the files. To input to this field you must create a logical file name (you can use ZARIBA_INVOICE_LOG_FNAME as it is defaulted) using the transaction FILE. In this transaction, you must specify a logical path and assign a physical file name. This physical file path should be less than 50 characters in length. /ARBA/CR_INVOICE_CREATE expects this input directory to contain subdirectories called Incoming, Processed, and Response. The Ariba Data Transfer Tool loads the data into the Incoming directory.

   This directory should match the %downloadDir%\%filePrefix% directory you specified in the Ariba Data Transfer Tool options file.

3. Specify the path directory separator character (\ or /).

4. To schedule /ARBA/CR_INVOICE_CREATE to run at regular intervals, save the program input as a variant that you can specify to be run as a job. Schedule this program using SAP transaction SM36. For more information about this transaction or scheduling SAP programs, see your SAP documentation.

Ensuring that shipping and special handling are included as charges

When exporting invoices to SAP ERP, if shipping and special handling were included as Charges, the export was failing in SAP ERP. This was happening for both file-based integration and web services-based integration. To ensure that the export is successful if shipping and special handling are included as Charges, a new field CHARGE_FLAG is introduced in the Ariba Procurement Solution. To ensure that invoices with shipping and special handling included as Charges are exported successfully, you must perform the following steps:

Prerequisites

To avoid receiving the error message Tax code does not exist for jurisdiction code, apply the SAP Note 1760355-EDI/BAPI: Unplanned delivery costs: Jurisdiction code.

1. Publish the new field CHARGE_FLAG in the Ariba Procurement Solution.

2. Import the latest ARIBA_ONDEMAND_INVOICE_CI_x_of_ariba.com.tpz file into your SAP Process Integration. You can download this file from https://connect.ariba.com. This step is applicable only for web services-based integration.

3. Import the relevant SAP transport request into your SAP ERP system. You can download the transport request from https://connect.ariba.com.
Accommodating custom fields

The /ARBA/CR_INVOICE_CREATE program reads the CSV files through CSV structures and internally calls the /ARBA/BAPI_INVOICE_CREATE RFC. If you want to add any custom/ additional fields, you must add the fields in the CSV structures. The following are extendable structures in RFC:

- **Header data**: /ARBA/INVOICE_CSV_HEADER
- **GL Account data**: /ARBA/INVOICE_CSV_LINE
- **Account data**: /ARBA/INVOICE_CSV_LINE
- **Item data**: /ARBA/INVOICE_CSV_ITEMDATA
- **Tax data**: /ARBA/INVOICE_CSV_TAX

The following are extendable structures in RFC:

- **Header data**: /ARBA/INVOICE_CREATE_HEADER
- **GL Account data**: /ARBA/INVOICE_CREATE_GLACCOUNT
- **Account data**: /ARBA/BAPI_INVOICE_CREATE
- **Item data**: /ARBA/INVOICE_CREATE_ITEM
- **Tax data**: BAPI_INCINV_CREATE_TAX

Calling the SAP RFC

The program /ARBA/CR_INVOICE_CREATE internally calls the RFC /ARBA/BAPI_INVOICE_CREATE, processes the data, and exports the invoice to SAP ERP through the BAPI.

Checking for duplicate invoices

Duplicate invoices are detected when there is an existing invoice with the same Ref_Doc_No (supplier invoice number), vendor, and fiscal year of the posting date. You can allow duplicate invoices by setting the parameter /ARBA/INVOICE_ALLOW_DUPS in the table /ARBA/TVARV. This table is the custom table for specifying parameter values for the integration function module.

Posting special charges

Header level shipping and special handling charges are posted as unplanned delivery costs in SAP ERP, if the SAP ERP configuration is set to Distribute among invoice line items for unplanned delivery costs. If this value is set to Different GL Line, the header level charges are posted to GL accounts sent from the Ariba Procurement Solution.
Jurisdiction codes

Jurisdiction code comes from the plant master and is placed in the GL account data table structure of the RFC. This is because the Ariba Procurement Solution does not maintain a jurisdiction code and it is not included in the CSV files. However, jurisdiction codes are required for tax calculations and tax postings in SAP ERP.

Time-out settings for invoice exports

Users can configure time-out settings for the Invoice export task to prevent invoices from being locked in the SAP ERP system. The RFC Connection time interval setting is maintained in seconds in the /ARBA/TVARV table.

You can specify the following column values in the /ARBA/TVARV table:

- **Variable Name** = /ARBA/INVOICE_PUSH
- **Selection Category** = P
- **INCL/EXCL** = I
- **Selection Value** = <Specify the SAP RFC Time Interval>

Scheduling

You should schedule the Ariba Data Transfer Tool frequently enough to keep up with new transactions. Use a third-party scheduler such as the one provided by your operating system. The frequency you set will vary depending on the nature of your business. You should schedule /ARBA/CR_INVOICE_CREATE to pick up new CSV files whenever they arrive.

Limitations for exporting invoices

Payments integration in this release has the following limitations:

- Ariba does not include mixed invoices in CSV files for SAP ERP. A mixed invoice is one that has both positive and negative amounts. For example, if there is a cash discount line, that is represented as a negative amount in the Ariba Procurement Solution.
- Invoices are sent to the SAP ERP after reconciliation in the Ariba Procurement Solution. Hence, these are reconciled Invoices sent to SAP ERP.
- The Ariba Procurement Solution sends Tax Code to SAP ERP. During invoice reconciliation, if you accept any tax variances, it can fail in the SAP ERP system based on the SAP ERP system set up.
- When you create an invoice for a PO which has a line item split into two line items, you must retain the same accounting information combination for each line item as was specified while creating the PO.
- An invoice cannot have only zero amount line items.
- If an invoice contains a mix of both zero and non-zero amount line items, invoices are created only for the non-zero amount line items.
- The OK-to-Pay Invoice Push feature applies only to Ariba Procure-to-Pay.
● PO based and non-PO based invoices are integrated to MM module of SAP ERP. Ariba provided ABAP programs ensure that all MM invoices create an FI document (accounting document) internally.
● There is no default contract integration with SAP ERP. Therefore, contract reference information is not sent to SAP ERP along with non-PO invoice integration. To do this, customers need to perform customizations accordingly.
● When you create a non-PO based invoice, Ariba Procurement Solution does not support Assets as an account assignment category.

Exporting expense reports

The SAP integration function module /ARBA/CR_ER_INVOICE_CREATE loads expense report CSV files into your SAP instance client as invoices. You can run /ARBA/CR_ER_INVOICE_CREATE manually or by scheduling it in SAP. The following topics explain /ARBA/CR_ER_INVOICE_CREATE:

• Understanding the expense report export [page 193]
• Running the expense report export [page 194]
• Accommodating custom fields [page 194]
• Calling the SAP BAPI [page 195]
• Scheduling [page 196]

Note

Expense reports are created as invoices in SAP ERP. For this reason you should create employees as suppliers. Load the supplier information for the employee into Ariba on-demand applications using the User.csv file (the SAPEmployeeSupplierId field). For more information on this CSV file, see the Ariba Travel and Expense Data Import Guide for SAP.

Understanding the expense report export

In general, the process works as follows:

• The Ariba Procurement Solution applications create the invoice data objects in the following CSV files:
  ○ SAPExpenseReportHeaderExport.csv.
  ○ SAPExpenseReportDistributionExport.csv
  ○ SAPExpenseReportTaxationExport.csv

For information on these files see the Ariba Travel and Expense Administration Guide.

• Run the Ariba Data Transfer Tool on the SAP server to make sure all the latest data is present in the CSV files.
• Run the SAP program /ARBA/CR_ER_INVOICE_CREATE. This pushes the Ariba expense report data in the CSV files into the SAP ERP database.
Running the expense report export

The input for this function module includes the directory where the CSV files should be written and the path separator character. Setting that up requires the following settings in the Ariba Data Transfer Tool options file for this download:

- Use the Ariba Data Transfer Tool Event Export Expense Reports.
- The downloadDir option must include \incoming at the end.
- The downloadDir + filePrefix options cannot exceed 75 characters.

How to run the invoice data push

Procedure

1. In SAP ERP, run the program /ARBA/CR_ER_INVOICE_CREATE using SAP transaction code SE38. This starts the SAP user interface for this program.
2. /ARBA/CR_ER_INVOICE_CREATE expects as input, a logical file path, which indicates where the Ariba Data Transfer Tool has downloaded the files. To input to this field, you must create a logical file name (you can use ZARIBA_INV_SSP_LOG_FNAME as it is defaulted) using the transaction FILE. In this transaction, you must specify a logical path and assign a physical file name. This physical file path should be less than 50 characters in length. /ARBA/CR_ER_INVOICE_CREATE expects this input directory to contain subdirectories called Incoming, Processed, and Error. The Ariba Data Transfer Tool loads the data into the Incoming directory. The specified parent directory must match the %downloadDir%\%filePrefix% directory you specified in the Ariba Data Transfer Tool options file that you are using to push expense report data to SAP.
3. Specify the path directory separator character (\ or /).
4. To schedule /ARBA/CR_ER_INVOICE_CREATE to run at regular intervals, save the program input as a variant that you can specify to run as a job. Schedule this program using SAP transaction SM36. For more information about this transaction or scheduling SAP programs, see your SAP documentation.

   i  Note
   The /ARBA/CR_ER_INVOICE_CREATE program has the following includes:

- /ARBA/CR_ER_ERRORHANDLING
- /ARBA/CR_ER_READERF01

Accommodating custom fields

The main function module is /ARBA/READ_ER_FILES, which reads all the folders in the parent directory. There are three BAdI methods so that you can use to extend the CSV file read and populate any additional fields.
The main function module uses these additional function modules:

- **RZL_READ_DIR** – This is a standard SAP function module that reads all the files.
- **/ARBA/KCD_CSV_FILE_CONVERSION** – This function module is copied from the standard SAP function module KCD_CSV_FILE_TO_INTERN_CONVERT. The difference is that /ARBA/KCD_CSV_FILE_CONVERSION calls the function module FILE_OPEN and the input parameter UPL = ' ', to read data from the application server. Input to /ARBA/KCD_CSV_FILE_CONVERSION is the file name and the file separator.

/ARBA/KCD_CSV_FILE_CONVERSION is called three times to read the three CSV files. It opens the CSV files and uses the data to populate the data internal table. The main table reads the internal table and populates the internal tables HEADERDATA, GLACCOUNTDATA, and TAXDATA which are in the Ariba structures /ARBA/ER_CSV_HEADER, /ARBA/ER_CSV_LINE, and /ARBA/ER_CSV_TAX.

If you have made any changes to the downloaded CSV files and want to get additional information into internal tables, add the additional fields to the structures /ARBA/ER_CSV_HEADER, /ARBA/ER_CSV_LINE, and /ARBA/ER_CSV_TAX.

If you would like to ask Ariba to consider adding a new field to a CSV file, call Ariba Customer Support to find out if this addition is feasible.

### Calling the SAP BAPI

The Ariba Procurement Solution uses /ARBA/ER_INVOICE_CREATE to create the expense data in SAP ERP. Before calling this BAPI, the function module code does some messaging or computations, as described in the following sections.

#### Determining document type

The document type that SAP ERP uses to create invoices for expense report integration is determined by the /ARBA/ER_DOC_TYPE parameter, which is set in the /ARBA/TVARV table. If there is no value there, the document type is set to ER.

#### Offset entry for company paid expense

The expense report could contain expenses that are paid by the company and are paid by the employee. In the distribution file, if the column EMP_PAID is NO and DB_CR_IND is S, then it is a company-paid expense and the employee will not get reimbursed. Therefore, you must add an additional line item with the posting for the offset/clearing account.

To determine the offset/clearing account there is a custom table /ARBA/ER_GL_DATA. This table has the following fields:

- **MANDT** – Client
- **BUKRS** – Company code
• SAKNR – G/L account

There is a table maintenance generated for this table and you can populate the values in this table by running the transaction code SM30.

**Calculating the gross amount**

The Ariba Procurement Solution integrated with SAP ERP does not consider the cash advance for two reasons:

- Cash advances are paid to employees before travel. There would be some accounting documents created before the payment is made to the employee.
- The cash advance on the T&E expense report is a single field that can just capture the amount but no accounting information related to it. Therefore, you cannot include this as-is amount line in the invoice.

For this reason, you must compute the gross amount in the invoice header. To compute this, the Ariba Procurement Solution adds the ITEM_AMOUNT on all the lines of the expense report, and adds the TAX_AMOUNT on each line to the ITEM_AMOUNT to arrive at the GROSS_AMOUNT.

**Changing INVOICE_IND**

The INVOICE_IND set in the CSV file is based on the GROSS_AMOUNT, which considers the CASH_ADVANCE. Therefore, in some cases it may indicate the credit memo, but with the CASH_ADVANCE taken into account, it may have to be created as an INVOICE. This is determined by the GROSS_AMOUNT computation mentioned above. If the GROSS_AMOUNT computed is:

- Positive, the Ariba Procurement Solution creates an INVOICE.
- Negative, the Ariba Procurement Solution creates a CREDIT_MEMO.

**Scheduling**

You should schedule the Ariba Data Transfer Tool frequently enough to keep up with new transactions. The frequency you set will vary depending on the nature of your business. You should schedule /ARBA/CR_ER_INVOICE_CREATE to pick up new CSV files whenever they arrive.

**Importing remittance advice**

The Ariba Procurement Solution loads remittance data from CSV files that the SAP integration program /ARBA/CR_REMITTANCE_EXPORT creates on your SAP instance client. This function module can import remittance advice for payments created for the following:

- Invoices
● Credit and debit memos
● Future-dated payments
● Voided payments, including voided checks and reverse payments

The following topics explain the remittance import process:

● Understanding the Remittance Data Import Event [page 197]
● Running the Remittance Data Import Event [page 197]
● Checking Output CSV Files [page 198]
● Additional Considerations for Remittance Data Import [page 199]
● Accommodating Custom Fields [page 199]

Understanding the remittance data import event

The following high-level steps describe the remittance data import process:

1. To process remittances, maintain the following payment document type entry in the /ARBA/TVARV table before running the program /ARBA/CR_REMITTANCE_EXPORT:
   ○ Variable Name: REMITTANCE_EXPORT
   ○ Field name: BLART
   ○ Selection cat.: P
   ○ Number: 0

2. Run the SAP program /ARBA/CR_REMITTANCE_EXPORT. This program reads from the table /arba/rem_datetm to get the last date time when this program was run, picks up all the payments done after the last run, and transfers the SAP ERP remittance data into the CSV files on the SAP server. The remittance data CSV files are then sent to the Ariba Procurement Solution.

How to run the remittance data import event

Context

The input for /ARBA/CR_REMITTANCE_EXPORT includes the directory where the CSV files should be written, the path separator character, and the character encoding to use.

 Procedure

1. In SAP ERP, run the function module /ARBA/CR_REMITTANCE_EXPORT using SAP transaction code SE38. This starts the SAP user interface for this function module.

2. Specify the directory path on the SAP application server in which to create the CSV files. This directory should already exist. For example c:\remittance\InDir. Be sure to use the same directory you specify in the Ariba Data Transfer Tool options file that you use when you import remittance data from SAP ERP. The directory should match $topDir%\$inDir%.
3. Specify the directory separator character (/ or \).

4. Specify the character encoding to use in the generated CSV files. Use the encoding that corresponds to the data you are supplying from SAP ERP. Ariba supports any of the encodings listed for the Java.io API at http://java.sun.com/j2se/1.4.2/docs/guide/intl/encoding.doc.html

5. To schedule \ARBA\CR_REMITTANCE_EXPORT to run at regular intervals, save the function module input as a variant that you can specify to be run as a job. Schedule the job using SAP transaction SM36. For more information about this transaction or scheduling SAP function modules, see your SAP documentation.

### Note

To import remittance advice from a specified date for the initial run, use SAP transaction SM30. Specify the table /arba/rem_datetm and click Maintain. Enter the start date/time as YYYYMMDDHHMMSS. When \ARBA\CR_REMITTANCE_EXPORT runs, it stores the run date/time in this table and uses it for the next run.

6. When writing the remittance data CSV files is complete:
   - Schedule the Ariba Data Transfer Tool to run event Import Remittance Data to load the CSV files each time \ARBA\CR_REMITTANCE_EXPORT has written them.
   - Make sure the Ariba Data Transfer Tool input directory specification matches the directory where you created the CSV files, as noted in step 2.

### Checking output CSV files

When the program \ARBA\CR_REMITTANCE_EXPORT has finished, you can check that the CSV files are generated as expected and they are located where the Ariba Data Transfer Tool can find them.

\ARBA\CR_REMITTANCE_EXPORT creates a temporary directory called processingyyyyymmddhhmmss, where yyyyymmddhhmmss is the time stamp indicating when the directory was created. It writes the Remittance.csv and RemittanceDetails.csv files into this directory. The Processing directory is placed inside the folder you specified as input in step 2, above.

### Note

In the Ariba Data Transfer Tool options file, leave the default value of “processing” as the filterPrefix value so that the Ariba Data Transfer Tool ignores this directory.

When \ARBA\CR_REMITTANCE_EXPORT is finished writing the files, it renames the temporary directory to processedyyyyymmddhhmmss. This directory is where the Ariba Data Transfer Tool looks for the CSV files. The next time the Ariba Data Transfer Tool runs, it finds this directory and transfers the data.

The \ARBA\CR_REMITTANCE_EXPORT program creates two files, Remittance.csv, and RemittanceDetails.csv.
Additional considerations for remittance data import

SAP ERP does not support remittance import in the following scenarios:

- The payment document contains withholding tax.
- The payment document has been reset.
- The payment document is an intercompany payment.
- The payment is for an invoice created for an expense report.
- The payment document is a down payment or a residual payment.

The following considerations pertain to remittance advice import:

- When you use the Check Payment method, remittance advice is imported only after the checks have been printed.
- If you make a payment to a special general ledger (G/L) account, remittance advice is not imported into your Ariba on-demand application. Common special G/L accounts might include:
  - Down payments
  - Bills of exchange
  - Guarantees
  - Security deposits
  - Individual value adjustments
  - Amortization
  - Interest payable
- The payment documents generated in SAP ERP from manual payment might not include the payment method by default as this is not a required field in SAP ERP. However, in Ariba Procurement Solution, the payment method is a required field and if it is empty in Remittance.csv, the load fails. To avoid a load failure, make sure you enter the payment method in the payment documents generated from a manual payment.
- The payment documents generated in SAP ERP from cancel payment might not include the payment method by default as this is not a required field in SAP ERP. However, in Ariba Procurement Solution, the payment method is a required field and if it is empty in Remittance.csv, the load fails. To avoid a load failure, when mapping the values in Ariba Procurement Solution, ensure that the Ariba Procurement Solution maps the payment method as C (Check) for cancel Payments, when there is no value set in SAP ERP.
- For manual payment, if you print the check after /ARBA/CR_REMITTANCE_EXPORT runs, then /ARBA/CR_REMITTANCE_EXPORT pulls the check number into Ariba on-demand application during the next remittance advice run and updates the Payment Number in the Payment Transaction object. However, the payment number is not pushed to the Ariba Network supplier.
- /ARBA/CR_REMITTANCE_EXPORT imports down payments as credit-line items after subsequent payments are made, not at the time they are created.

Accommodating custom fields

The custom user BAdI methods /ARBA/CR_REMITTANCE_EXPORT is provided so that you can add or change fields in the transferred data. Restricting your customizations to custom user BAdI methods will help you integrate new versions of the Ariba SAP integration function modules.
Integrating transactional data using the web services channel

About transaction data [page 200]
Components of an integration event [page 203]
Integrating purchase orders [page 207]
Integrating amount-based purchase orders [page 208]
Integrating changed and canceled purchase orders [page 209]
Integrating closed purchase orders [page 210]
Integrating receipts [page 212]
Integrating payments [page 218]
Integrating remittance [page 219]
Integrating real-time budget check [page 221]
Integrating taxes and charges [page 227]
Integrating advance payments [page 230]
Integrating non-PO (FI) invoices [page 232]
Integrating service procurement [page 234]
Integrating attachments support [page 237]
How to customize an existing integration event [page 242]

About transaction data

Transaction data includes information like remittance advice, invoice, purchase orders, and receipts. Export events, when using the web services channel occur in real-time (triggered when the transactions are ready to be sent to the SAP ERP).

Import events such as a Remittance Import are scheduled tasks.
Typical integration event process flow

The following illustration shows a typical data integration process flow using web services:

![Figure 8: Typical Data Integration Process Flow Using Web Services](image)

Understanding SOAP URLs

The SOAP URL is the endpoint on the Web Service. For outbound events (where performs the request: Request-Reply and Publish), this SOAP URL needs to be explicitly configured through Ariba Administrator.

For outbound events (where the external EAI system performs the request: Service and Subscribe), the SOAP URL is generated automatically to be present in the generated WSDL according to the following logic:

```
<IncomingHttpServerURL> / <ContextRoot> / soap / <realm name> / <event_name>
```

where:
- `IncomingHttpServerURL`, `ContextRoot`: are taken from the AppInfo.xml file located in `<install>/config/asmshared`
- `soap`: is the static alias for the Web Service Servlet intended to process the WS requests
- `soap_servlet_alias`: is the alias used for the WS channel servlet
- `partition`: is the name of the partition
- `event_name`: is the name of the event

Consider the following AppInfo.xml example, where the partition is `psap` and the topic name is `PurchaseOrderExport`:

```xml
<AppInfo Version="3.0">
  <Instance isCDS="true" name="Buyer">
    <Param name="ContextRoot" value="Buyer"/>
    <Param name="IncomingHttpServerURL" value="https://buyer.ariba.com:4443"/>
    [...]</Instance>
</AppInfo>
```

The generated SOAP URL will be:

https://buyer.ariba.com:4443/Buyer/soap/psap/PurchaseOrderExport

The value of the SOAP URL for the inbound events can be configured through the Ariba Administrator.
Understanding WSDLs

WSDL is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. It provides a model and an XML format for describing Web services.

The abstract definition of ports and messages is separated from their concrete use or instance, allowing the reuse of these definitions. A port is defined by associating a network address with a reusable binding, and a collection of ports define a service. Messages are abstract descriptions of the data being exchanged, and port types are abstract collections of supported operations. The concrete protocol and data format specifications for a particular port type constitutes a reusable binding, where the operations and messages are then bound to a concrete network protocol and message format. In this way, WSDL describes the public interface to the web service.

WSDL is often used in combination with SOAP and XML Schema to provide web services over the Internet. A client program connecting to a web service can read the WSDL to determine what functions are available on the server. Any special data types used are embedded in the WSDL file in the form of XML Schema. The client can then use SOAP to actually call one of the functions listed in the WSDL.

A WSDL is made up of the following sections:

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>import</td>
<td>This element is used to associate a namespace with a document location. We do not currently use this tag.</td>
</tr>
<tr>
<td>types</td>
<td>This element is used to define user created data types which will be used in the document.</td>
</tr>
<tr>
<td>message</td>
<td>This element defines all the parts of an individual message.</td>
</tr>
<tr>
<td>portType</td>
<td>This is a container of supported operations by the web service. The operations in the portType are ordered and indicate whether a message is inbound or outbound.</td>
</tr>
<tr>
<td>binding</td>
<td>This element defines the operation to protocol mapping. (for example, http, https, MIME, etc.).</td>
</tr>
<tr>
<td>service</td>
<td>This element defines the operation to address mapping. So this is the actual address the request should be sent to.</td>
</tr>
</tbody>
</table>

Ariba provides vsap as the default target namespace in the WSDL. You must convert your target namespace to vsap namespace, otherwise all transactions fail with an error message, for example:

```
EXCEPTION_DURING_EXECUTE
com/sap/xi/tf/ MM_WSPurchaseOrderPush_RequestMessage~
com.sap.aii.mappingtool.tf7.illegalInstanceException: Cannot create target element /ns1:_ARBA_-BAPI_PO_CREATE1/PARTITION.
Values missing in queue co~
Runtime exception occurred during application mapping
com/sap/xi/tf/ MM_WSPurchaseOrderPush_RequestMessage~
com.sap.aii.mappingtool.tf7.illegalInstanceException: Cannot create target element /ns1:_ARBA_-BAPI_PO_CREATE1/PARTITION.
Values missing in queue co~
```

To convert your target namespace to vsap namespace, contact your Ariba Support Representative.
## Additional References

For more information about how to configure Web Services events from the Ariba Administrator, see the *Ariba Procurement Data Import and Administration Guide*.

## Components of an integration event

The following table lists the various components that constitute every transactional data integration event. These components are spread across both Ariba Procurement Solution and SAP ERP.

<table>
<thead>
<tr>
<th>Transactional Data</th>
<th>Event Name</th>
<th>SWC Name and NameSpace</th>
<th>RFC/IDoc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requisition</td>
<td>RequisitionRealTimeBudgetExport</td>
<td>Ariba_ONDEMAND_BUDGET_MGMT CIx of ariba.com</td>
<td>/ARBA/PURREQ</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(urn:ariba.com:xi:OnDemand:PurchaseRequisition)</td>
<td></td>
</tr>
<tr>
<td>Delete Requisition</td>
<td>RequisitionRealTimeRevertBudgetExport</td>
<td>Ariba_ONDEMAND_BUDGET_MGMT CIx of ariba.com</td>
<td>/ARBA/PURREQ_DELETE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(urn:ariba.com:xi:OnDemand:PurchaseRequisition)</td>
<td></td>
</tr>
<tr>
<td>Purchase Order</td>
<td>PurchaseOrderExport</td>
<td>Ariba_ONDEMAND_PURCHASE_ORDER, CIx of ariba.com</td>
<td>/ARBA/BAPI_PO_CREATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(urn:ariba.com:xi:OnDemand:CreatePurchaseOrder)</td>
<td></td>
</tr>
<tr>
<td>Changed Purchase Order</td>
<td>PurchaseOrderChangeExport</td>
<td>-do- (urn:ariba.com:xi:OnDemand:ChangePurchaseOrder)</td>
<td>/ARBA/BAPI_PO_CHANGE</td>
</tr>
<tr>
<td>Canceled Purchase Order</td>
<td>PurchaseOrderCancelExport</td>
<td>-do- (urn:ariba.com:xi:OnDemand:CancelPurchaseOrder)</td>
<td>/ARBA/BAPI_PO_CANCEL</td>
</tr>
<tr>
<td>Purchase Order Header Status</td>
<td>PurchaseOrderHeaderStatusImport</td>
<td>-do- (urn:ariba.com:xi:OnDemand:PurchaseOrderHeaderStatus)</td>
<td>/ARBA/PO_HEADER_STATUS</td>
</tr>
<tr>
<td>Receipts</td>
<td>ReceiptExport</td>
<td>Ariba_ONDEMAND_RECEIPT, CIx of ariba.com</td>
<td>/ARBA/GOODS_RECEIPTS_CREATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(urn:ariba.com:xi:OnDemand:Receipt)</td>
<td></td>
</tr>
</tbody>
</table>
The following table lists the various components that constitute every functional data integration event. These components are spread across both Ariba Procurement Solution and SAP ERP.

<table>
<thead>
<tr>
<th>Functional Data</th>
<th>Event Name</th>
<th>SWC Name and NameSpace</th>
<th>RFC/IDoc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invoice</td>
<td>PaymentExport</td>
<td>ARIBA_ONDEMAND_INVOICE, CIx of ariba.com (urn:ariba.com:xi:OnDemand:Invoice)</td>
<td>/ARBA/BAPI_INVOICE_CREATE</td>
</tr>
<tr>
<td>Remittance (Is a scheduled task)</td>
<td>RemittanceImport</td>
<td>ARIBA_ONDEMAND_REMITTANCE, CIx of ariba.com (urn:ariba.com:xi:OnDemand:Remittance)</td>
<td>/ARBA/CR_REMITTANCE_EXPORT</td>
</tr>
<tr>
<td>Advance Payment</td>
<td>• ImportAdvancePayment&lt;br&gt;• ErrorsImportAdvancePayment ID&lt;br&gt;• ExportAdvancePayments</td>
<td>ARIBA_ONDEMAND_ADVANCE_PYMNT, CIx of ariba.com (urn:ariba.com:xi:OnDemand:AdvancePayment)</td>
<td>/ARBA/ADV_PAYMENT_POST</td>
</tr>
<tr>
<td>Cancel Advance Payment</td>
<td>• ImportCancelAdvancePaymentID&lt;br&gt;• ExportCancelAdvancePayments</td>
<td>ARIBA_ONDEMAND_ADVANCE_PYMNT, CIx of ariba.com (urn:ariba.com:xi:OnDemand:AdvancePayment)</td>
<td>/ARBA/ADV_PAYMENT_REV_POST</td>
</tr>
<tr>
<td>Advance Payment Remittance</td>
<td>Import Advance Payment Remittance</td>
<td>ARIBA_ONDEMAND_ADVANCE_PYMNT, CIx of ariba.com (urn:ariba.com:xi:OnDemand:AdvancePaymentRemittance)</td>
<td>/ARBA/ADV_PAY_REMIT_EXPORT</td>
</tr>
<tr>
<td>Service Entry Sheet</td>
<td>Export Service Sheets to External System</td>
<td>ARIBA_ONDEMAND_SRVENTRY-SHEET, CIx of ariba.com (urn:ariba.com:xi:OnDemand:CreateServiceEntrySheet)</td>
<td>/ARBA/ENTRYSHEET_CREATE</td>
</tr>
<tr>
<td>Service Entry Sheet Response</td>
<td>Import Service Sheet Approval Statuses</td>
<td>ARIBA_ONDEMAND_SRVENTRY-SHEET, CIx of ariba.com (urn:ariba.com:xi:OnDemand:ServiceEntrySheetResponse)</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Budget Check</td>
<td>RequisitionBudgetCheckExport</td>
<td>ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com (urn:ariba.com:xi:OnDemand:BudgetCheck)</td>
<td>/ARBA/ACC_PURCHASE_REQUI_CHECK</td>
</tr>
</tbody>
</table>
Functional Data | Event Name | SWC Name and NameSpace | RFC/IDoc
---|---|---|---
Funds Derivation:  
  ● Requisition Derive  
  ● Invoice Derive | RequisitionDerive AccountingExport  
InvoiceReconciliationDeriveAccountingExport | ARIBA_ONDEMAND_BUDGET_MGMT CIx of ariba.com  
(urn:ariba.com:xi:OnDemand:Derivation) | /ARBA/DERIVATION

The sample URLs in the Ariba Administrator under **Integration Manager** ➔ **End Point Configuration** ➔ for each data integration event is listed here:

<table>
<thead>
<tr>
<th>Event Name</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>PurchaseRequisitionExport</td>
<td><a href="http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_Procurement:CC_ReqSubmit_Sender">http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_Procurement:CC_ReqSubmit_Sender</a></td>
</tr>
<tr>
<td>PurchaseOrderHeaderStatusImport</td>
<td><a href="http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_Procurement:CC_PurchaseOrderHeader_WSDL_Sender">http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_Procurement:CC_PurchaseOrderHeader_WSDL_Sender</a></td>
</tr>
<tr>
<td>ReceiptExport</td>
<td><a href="http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_Procurement:CC_Receipts_WSDL_Sender">http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_Procurement:CC_Receipts_WSDL_Sender</a></td>
</tr>
<tr>
<td>PaymentExport</td>
<td><a href="http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_Procurement:CC_Invoice_WSDL_Sender">http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_Procurement:CC_Invoice_WSDL_Sender</a></td>
</tr>
<tr>
<td>RemittanceImport</td>
<td><a href="http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_Procurement:CC_RemittanceImport_WSDL_Sender">http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_Procurement:CC_RemittanceImport_WSDL_Sender</a></td>
</tr>
<tr>
<td>AdvancePaymentCreate</td>
<td><a href="http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_ONDEMAND:CC_Adv_PaymentRequest_Create_Sender">http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_ONDEMAND:CC_Adv_PaymentRequest_Create_Sender</a></td>
</tr>
<tr>
<td>AdvancePaymentCancel</td>
<td><a href="http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_ONDEMAND:CC_Adv_PaymentRequest_Cancel_Sender">http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_ONDEMAND:CC_Adv_PaymentRequest_Cancel_Sender</a></td>
</tr>
<tr>
<td>AdvancePaymentRequestRemittance</td>
<td><a href="http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_ONDEMAND:CCT_Adv_PaymentRequestRemittance_WSDL_Sender">http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_ONDEMAND:CCT_Adv_PaymentRequestRemittance_WSDL_Sender</a></td>
</tr>
<tr>
<td>ServiceEntrySheet</td>
<td><a href="http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_ONDEMAND:CCT_ServiceEntrySheet_WSDL_Sender">http://host:port/XISOAPAdapter/MessageServlet?channel=:BS_ONDEMAND:CCT_ServiceEntrySheet_WSDL_Sender</a></td>
</tr>
</tbody>
</table>
Advanced pricing details

While creating a requisition in the Ariba Procurement Solution, the user can specify advanced pricing details for a line item or multiple line items. Advanced pricing details includes:

- Quantity Based Pricing
- Unit Conversion Pricing

Quantity based pricing allows the unit price of an item to be based on a different price unit quantity than 1. Unit Conversion Pricing allows unit of measure conversion in the pricing calculation, when the unit of measure on the order differs from the pricing unit of measure.

While creating a requisition in the Ariba Procurement Solution, if the user has specified advanced pricing details for a line item, then in the exported SOAP message, the following fields are added:

- PriceBasisQuantity
- PriceBasisQuantityUOM
- ConversionFactor
- PriceBasisQuantityDesc

Advanced pricing details are also applicable for change purchase order and payment requests. Therefore, when a change PO and payment request are exported, the above mentioned fields are available in the respective SOAP message.

To ensure that the above mentioned fields related to advanced pricing details are processed in SAP Process Integration and your SAP ERP system, download the relevant TPZ files and SAP transport requests from connect.ariba.com and import them into SAP Process Integration and SAP ERP system respectively.

Scheduled task for PurchaseOrdHeaderStatusImport

The PurchaseOrderHeaderStatusImport integration event can also be scheduled by using the ProcessERPHeaderStatusPull scheduled task. An example of the scheduled task is as follows:

```
ProcessERPHeaderStatusPull = {
    QueryPeriod = 30;
    ScheduledTaskClassName = "ariba.purchasing.ordering.ProcessERPHeaderStatusPull";
    Schedules = { Schedule1 = { DayOfWeek = Everyday; Hour = 1;};};
};
```

This schedule tasks picks up all SAP ERP Orders with the Ordered status that have been created in the last 30 days (QueryPeriod), and exports them to SAP ERP through SAP Process Integration.

SAP ERP returns values such as ERPAllowCancel to be True or False (that determines whether or not to allow the Cancel PO action).
Scheduled task for RemittanceImport

The RemittanceImport integration event can also be scheduled using the IncrementalRemittanceImport scheduled task. An example of the scheduled task is as follows:

```
IncrementalRemittanceImport = {
    DefaultStartDate = 20080901000000;
    Event = RemittanceImport;
    LoggingName = IncrementalRemittanceImport;
    Partition = psap;
    ScheduledTaskClassName = "ariba.integration.core.IncrementalIntegrationEventTask";
    Variant = vsap;
    Schedules = { Schedule1 = { DayOfWeek = Everyday; Hour = 4;};};
};
```

**Note**
The DefaultStartDate parameter can be used as shown in the above example, to import the remittance documents from a particular date and time when running event for the very first time.

Subsequent runs will be incremental and the start date submitted in the event request will be the end date of the previously executed instance of the event.

Scheduled task for ReceiptExport

The ReceiptExport integration event can also be scheduled by using the ProcessPendingReceipts scheduled task. An example of the scheduled task is as follows:

```
ProcessPendingReceipts = {
    ProcessPeriod = 1;
    ScheduledTaskClassName = ariba.receiving.ProcessPendingReceipts;
    Schedules = { Schedule1 = { DayOfWeek = Weekday; Hour = 0;};};
};
```

**Note**
The ProcessPeriod parameter value indicates the lag in the number of days before the receipts are exported to SAP ERP when the scheduled task is executed.

If you want the scheduled task to pick up the receipts created on the current day, the ProcessPeriod parameter value needs to be changed to 0 (zero.)

Integrating purchase orders

An Ariba Procurement Solution purchase order is exported to SAP ERP as soon as it is fully approved in the Ariba Procurement Solution.
The following steps explain how a purchase order export integration event works:

1. Create a purchase order (PO) with multiple line items in. After the Requisition is fully approved, an SAP ERP Order Object is created and sent to SAP Process Integration as per the WSDL.

   **Note**

   When creating the PO, the *NeedByDate* field is not mandatory. However, when you push the PO to SAP ERP, this field is required else the PO push fails.

2. SAP Process Integration transmits the XML messages to the RFC Adapter, which process data and sends the response back, either as a Success (Number Import) or Error (Error Import). If the ERP Order is created successfully in SAP ERP, you receive the PO number from the SAP ERP, and the order status changes from *Ordering* to *Ordered*. If there is an error in the event, the status changes to *Composing*.

3. Update the Requisition Object with the ERP Order ID. The Requisition status changes to *Ordered*.

The following features are not provided in the export purchase order integration event:

- Exporting the PO titles to SAP ERP.
- Support for the *deliver to* or *ship to* address in an Ariba Procurement Solution PO. In many circumstances, the plant field provides the information for SAP ERP to deliver to the address field in PO.

**Related Information**

- Integrating real-time budget check [page 221]
- Integrating taxes and charges [page 227]

**Understanding BAPI import limitations**

The BAPI integration with the Ariba Procurement Solution does not provide support for the following features:

- Exporting tax code information
- The Ariba Procurement Solution exports field mapping mechanism. If you need to add the Ariba Procurement Solution extrinsic fields to RFC objects, you add these fields to the RFC structures. Next, you add this field in the AML mapping, re-import the schema into SAP Process Integration and make appropriate changes in the mapping.

If a purchase order or a change purchase order is exported using a RFC, the carriage returns in the line item text are exported and displayed as the hash character (#) in the line item text in the SAP ERP purchase order.

**Integrating amount-based purchase orders**

The Ariba Procurement Solution allows you to export amount-based purchase orders to SAP ERP. For amount-based purchase orders, the Ariba Procurement Solution maps the amount to the quantity in SAP ERP, and sets the unit price to 1. For example, if the amount of the purchase order in the Ariba Procurement Solution is $95, it is exported to SAP ERP as a receipt with a quantity of 95 and with a price of $1.
The Ariba Procurement Solution does not support exporting zero value amount for line items in amount-based purchase orders.

Integrating changed and canceled purchase orders

The Ariba Procurement Solution can export changed and canceled purchase orders (CPO) to SAP ERP. For information on this feature and how to enable it, see the chapter on changing existing orders in the Ariba Procurement Data Import and Administration Guide.

When the Ariba Procurement Solution exports a changed or canceled purchase order, it uses the RFC /ARBA/BAPI_PO_CHANGE or /ARBA/BAPI_PO_CANCEL to fill the RFC tables, and, in turn, calls BAPI_PO_CHANGE to export the purchase order to SAP ERP.

The following are important notes on exporting changed and canceled purchase orders to SAP ERP:

- If you enable the change/cancel purchase order feature, users must always change and cancel orders through the Ariba Procurement Solution and not change or cancel them through SAP ERP. Therefore, apply the appropriate authorization concept in SAP ERP to prevent users from changing purchase orders directly in SAP ERP.
- To enable the change/cancel purchase order feature, you install the change/cancel order transport. The transport number is listed in the transport readme.txt file. For more information see the Installation Guide for Ariba Procurement Solution Integrated with SAP ERP.
- A limitation in the standard SAP BAPI creates the following situation: If the Account Assignment category is modified in the Ariba Procurement Solution, the Account Assignment category is changed in SAP ERP. Furthermore, the GLAccount is changed to a default GLAccount after it has been pushed to SAP ERP. To resolve this issue, modify the GLAccount in SAP ERP using the Account Assignment tab in the Item Detail section.
- Plant data is not editable if the Purchase Order Line Item is partially or fully received or partially or fully invoiced in the Ariba Procurement Solution or SAP ERP.

Related Information

Integrating real-time budget check [page 221]
Integrating taxes and charges [page 227]

Adjusting net price

When exporting amounts, the Ariba Procurement Solution RFC adjusts the net price to the number of decimals specified for that currency in the SAP TCURX table.

If the currency is listed and specifies a number greater than zero in the Decimals column, the RFC adjusts the net price to the number of decimals and price unit accordingly.
• If the currency is not listed in the table, the RFC adjusts the net price to two decimal places, which is the default specified in the RFC.
• If the currency is listed in the table, but specifies zero in the Decimals column, the lengths of the net price and price unit are adjusted accordingly.
• Price units are adjusted to a maximum of 10,000 units, at which point, an error message is sent.
• In a changed purchase order, you cannot export a zero value amount in the Net Price for line items. For example, if the number of decimals allowed in SAP ERP is two for USD, and if the Ariba Procurement Solution sends a net price (price unit) of 100.023, the wrapper RFC adjusts the net price to 1000.23 and the price unit to 10. If the price unit exceeds 10000, an error message is sent to the Ariba Procurement Solution if PO Push using RFC is in use.

**Importing header status of purchase orders**

When changed orders are in a Received, Receiving, Invoiced, or Invoicing state, the Ariba Procurement Solution cannot cancel them. Similarly, when purchase orders have been received or invoiced in SAP ERP, the cancel export to SAP ERP fails.

If an order reaches SAP ERP successfully, its status can still change in SAP ERP. For example, if you receive against an order in SAP ERP, a subsequent cancel order from the Ariba Procurement Solution is certain to be rejected. To reduce the occurrence of such situations, the scheduled task ProcessERPHeaderStatusPull imports header status information from SAP ERP. This task sets a field in the Ariba Procurement Solution, ERPAllowCancel, which specifies whether the order can be canceled.

**Note**

Use the task ProcessERPHeaderStatusPull only if you have enabled change and cancel order exports to SAP ERP, and you are doing a lot of receiving in SAP ERP.

The scheduled task ProcessERPHeaderStatusPull does not eliminate all the cancel export failure errors. SAP ERP uses many factors to determine if it is possible to cancel an order. However, the status import is based on invoicing and receiving status.

The scheduled task ProcessERPHeaderStatusPull imports the status for all orders in the Ordered state for the time period (days before the current date) you specify in the QueryPeriod parameter.

You can also specify the schedule for the task ProcessERPHeaderStatusPull in the ScheduledTask.table. The RFC /ARBA/PO_HEADER_STATUS is used to import the status information from SAP ERP.

**Integrating closed purchase orders**

Ariba Cloud Integration Release 9.0 enables buyers using Ariba Procurement Solution integrated with SAP ERP to integrate purchase order closure information from Ariba Procurement Solution with SAP ERP.

Integration of purchase order closure information between Ariba Procurement Solution and SAP ERP ensures that the purchase order information in Ariba Procurement Solution and SAP ERP is in sync, and improves the fund allocation and utilization. When a purchase order in SAP ERP is marked as closed, SAP ERP releases any unused fund that was allocated for the purchase order and sets any open quantity to zero.
However, note that the integration also updates status of corresponding purchase orders in SAP ERP when buyers reopen purchase orders in Ariba Procurement Solution. If a buyer reopen a closed purchase order in Ariba Procurement Solution, SAP ERP returns any fund that was released as unused when the purchase order was closed.

In web service-based integrations, Ariba Procurement Solution sends the purchase order closure information to SAP ERP in real time. When you close a purchase order in Ariba Procurement Solution, SAP ERP receives a SOAP message with the details of the purchase order and the reasons for closure.

In Ariba Procurement Solution, buyers can specify a status while they close a purchase order. The following table lists the status messages that buyers can set in Ariba Procurement Solution and the corresponding status update in SAP ERP:

<table>
<thead>
<tr>
<th>Status in Ariba Procurement Solution</th>
<th>Status in SAP ERP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order Closed for Receiving</td>
<td>Delivery Complete</td>
</tr>
<tr>
<td>Order Closed for Invoicing</td>
<td>Final Invoice</td>
</tr>
<tr>
<td>Order Closed for All Actions</td>
<td>Delivery Complete and Final Invoice</td>
</tr>
</tbody>
</table>

**Note**

SAP ERP does not support the Order Closed and Adjusted for Receiving and Closed for Change status for purchase orders. Purchase orders closed with either of these status messages in Ariba Procurement Solution are not updated in SAP ERP.

When buyers reopen purchase orders that were previously-closed in Ariba Procurement Solution, the status check boxes, such as Delivery Complete or Final Invoice or both, in SAP ERP for the corresponding purchase orders are also cleared.

**Prerequisites**

- Install the latest Ariba components.
- Configure the purchase order integration and change purchase order integration between Ariba Procurement Solution and SAP ERP.
- Enable the following parameters in Ariba Procurement Solution:
  - Application.Purchasing.PullCloseOrderStatusAckFromERP
  - Application.Purchasing.PushCloseOrderStatusToERP

**Limitations**

- SAP ERP does not support the Order Closed and Adjusted for Receiving and Closed for Change status for purchase orders. Purchase orders closed with either of these status flags in Ariba Procurement Solution are not updated in SAP ERP.
SAP ERP does not support the Delivery Complete status for service purchase orders. Because of this limitation, SAP ERP does not update service purchase orders closed in Ariba Procurement Solution with the status Order Closed for Receiving. Similarly, for service purchase orders that are closed in Ariba Procurement Solution with the status Order Closed for All Actions, only the Final Invoice check box is checked in SAP ERP.

Integrating receipts

A receipt is an acknowledgment of the goods that have arrived. The receiving process starts when an order has been sent to a supplier and the supplier ships goods in exchange. When the shipment arrives, the person who receives those items, submits a receipt to acknowledge that the items have indeed arrived. Ariba provided integration code integrates receipts as goods receipts with Movement Type as 101.

You may also need to create a negative receipt in the Ariba Procurement Solution if you want to reverse some received quantity as a result of incorrect shipment or manual errors. Ariba provided integration code integrates negative receipts as goods receipts with Movement Type as 102.

The following steps explain how a receipt export integration event works:

1. When an item is received in the Ariba Procurement Solution, a receipt object instance is approved.
2. When the scheduled receipt export scheduled task runs, the Ariba Procurement Solution sends the receipt object to the SAP Process Integration.

   Note
   The scheduled task ProcessRendingReceipts posts receipts from Ariba Procurement Solution to SAP ERP. However, you cannot post a receipt successfully in SAP ERP if the period has been closed in SAP ERP.

   The following are ways to address this problem:
   The delay period of the scheduled task ProcessPendingReceipts must be set to ensure the receipt is approved in the Ariba Procurement Solution and, in turn, is posted in SAP ERP on the same day.
   Postpone the closing period in SAP ERP for one or two days so that when you post the receipt in SAP ERP, the period is still open.
   For more information on scheduled tasks, see the Ariba Procurement Data Import and Administration Guide.
3. The SAP Process Integration mapping defines the transformation of the data into the format required by the RFC. Then SAP Process Integration sends the data to the Ariba Procurement Solution receipt export RFCs through the RFC Adapter for SAP ERP.
4. The Ariba Procurement Solution /ARBA/GOODS_RECEIPTS_CREATE RFC processes the data and inserts it in receipt creation screens in SAP ERP.
5. The RFC returns a receipt line number and any error information to the Ariba Procurement Solution.

Related Information

Integrating receipts - Cloud Integration release 6.0 onwards [page 213]
Integrating receipts - Cloud Integration release 6.0 onwards

Starting from Cloud Integration release 6.0, receipts are created using a standard SAP delivered BAPI, BAPI_GOODSMVT_CREATE which provides following enhanced receipt functionalities:

- Fields customization using the BAPI parameter options
- Multiple account assignments for purchase orders

In addition, this feature reduces implementation efforts and avoids disruption in case of future SAP ERP upgrades.

The following sections apply if you are using Cloud Integration release 6.0:

- Functional changes [page 213]
- Mandatory fields for creating receipts [page 215]
- Migrating customization for receipts [page 216]
- About filtering and BAdI methods [page 216]
- Extension structures for export tasks [page 217]

Related Information

- Time-out Settings for Receipt Exports [page 217]
- Exporting Amount-Based Receipts [page 217]
- Limitations for Integrating Receipts [page 217]

Additional References

For more information pertaining to BAPI, see the BAPI_GOODSMVT_CREATE documentation [external document].

Functional changes

Buyers using the Ariba Procurement Solution integrated with SAP ERP can create receipts using a standard SAP delivered BAPI, BAPI_GOODSMVT_CREATE instead of using the batch input technique.

The following table describes the functional changes implemented to support BAPI_GOODSMVT_CREATE:
<table>
<thead>
<tr>
<th>Functional area</th>
<th>SAP object name</th>
<th>Change description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goods Receipt Program</td>
<td>/ARBA/CR_GR_CREATE</td>
<td>The calls to RFC function module /ARBA/GOODS_RECEIPTS_CREATE is updated with a new signature to support the BAPI. The error handling logic has been updated to accommodate the field changes in /ARBA/GOODS_RECEIPTS_ERROR structure.</td>
</tr>
<tr>
<td>Remote Function Module</td>
<td>/ARBA/GOODS_RECEIPTS_CREATE</td>
<td>To replace the batch input logic with BAPI:                                                                                           • Deleted  ○ The BDC codes from the RFC function module /ARBA/GOODS_RECEIPTS_CREATE. ○ Calls to BAdI method GOODS_RECEIPTS_BDCDATA.  • Modified  ○ /ARBA/GOODS_RECEIPTS_ERROR structure to accommodate the BAPI return errors and display the same in the new report. ○ Call transaction to MB01 replaced with BAPI_GOODSMVT_CREATE in the function module /ARBA/GOODS_RECEIPTS_CREATE.</td>
</tr>
<tr>
<td>Business Add-ins</td>
<td>/ARBA/GR_CREATE</td>
<td>• Deleted the BAdI method GOODS_RECEIPTS_BDCDATA from /ARBA/GR_CREATE.  • Created a new method in existing BADI /ARBA/GR_CREATE as PUBLISH_GR_CREATE with an option for customer enhancements.  • Exposed all the importing and tables parameters of BAPI_GOODSMVT_CREATE excluding BAPIRET2 structure for enhancement. The new method PUBLISH_GR_CREATE consists of parameters that can be customized and are passed to BAPI_GOODSMVT_CREATE.</td>
</tr>
</tbody>
</table>
### Functional area | SAP object name | Change description
--- | --- | ---
Structure | /ARBA/GOODS_RECEIPTS_ERROR | In the /ARBA/GOODS_RECEIPTS_ERROR structure:
- The BDC screen fields have been replaced with BAPI return fields to display errors in the new report.
- The earlier embedded structure /ARBA/ERROR_MESSAGE_CORE has been deleted. The fields used for the present interface are directly included in the /ARBA/GOODS_RECEIPTS_ERROR structure.

**Note**

For MSGID, the field length is now increased from 2 characters to 20 characters.

### Data Dictionary Objects

The following table lists the structures created and provided as modifiable parameters in the method BADI /ARBA/GR_CREATE~PUBLISH_GR_CREATE:

<table>
<thead>
<tr>
<th>Structure</th>
<th>Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/GOODS_RECEIPTS_HEADER</td>
<td>Structure embeds BAPI2017_GM_HEAD_01 (Material Document Header Data from: BAPI_GOODSMVT_CREATE)</td>
</tr>
<tr>
<td>/ARBA/GOODS_RECEIPTS_ITEM</td>
<td>Structure embeds BAPI2017_GM_ITEM_CREATE (BAPI Communication Structure: Create Material Document Item)</td>
</tr>
</tbody>
</table>

**Note**

To enhance customization options, the structures and table types are created in /ARBA namespace.

### Mandatory fields for creating receipts

To create receipts in SAP ERP, the following mandatory fields that are required to be populated in the BAPI_GOODSMVT_CREATE are available and mapped, in the Ariba Procurement Solution integrated with SAP ERP:

- GOODSMVT_HEADER
  - Posting Date
The RFC function module /ARBA/GOODS_RECEIPTS_CREATE signature has been enhanced with the BAPI parameter options, Import and Tables, and is accessible for customization.

Migrating customization for receipts

Buyers upgrading from an earlier version to Cloud Integration release 6.0 or higher are required to manually migrate receipts customization, if any, from the previous versions which used batch input technique.

After the upgrade, buyers can update the customization in the following BADI /ARBA/GR_CREATE methods:

- GOODS_RECEIPTS_CREATE
- GOODS_RECEIPTS_ERROR
- PUBLISH_GR_CREATE

About filtering and BAdI methods

The goods receipt RFCs offer several BAdI methods that allow you to filter the data they export. The BAdI methods work the same for exports as they do for imports—the only difference is the direction in which the data is moving.

For example, the goods receipt export RFCs offer several BAdI methods, which execute in the following order:

- BAdI method before any data processing begins:
  GOODS_RECEIPTS>Create
- BAdI method consisting of parameters that can be customized:
  PUBLISH_GR_CREATE
- BAdI methods at the end for testing error conditions:
  GOODS_RECEIPTS_ERRORS

Use these BAdI methods to manipulate goods receipt pushes in ways that field mappings do not offer.
Extension structures for export tasks

The Ariba Procurement Solution export RFCs provide extension structures that work the same as the extension structures for imports. The extension structures allow you to add fields or columns to the data that the Ariba Procurement Solution sends to SAP ERP.

For example, goods receipt export RFC, /ARBA/GOODS_RECEIPTS_CREATE, provides the following extension structures:

- /ARBA/GOODS_RECEIPTS_ITEM - for line item receiving information
- /ARBA/GOODS_RECEIPTS_HEADER - for header-level receiving information
- /ARBA/GOODS_RECEIPTS_ERROR - for error messages the RFC sends to the Ariba Procurement Solution (not part of the export)

Make sure that the corresponding worksheets set values for any fields you add to the extension structures.

Time-out Settings for Receipt Exports

The RFC Connection Time Interval is maintained in the /ARBA/TVARV table as an entry in seconds. This time interval must be the SAP RFC connection time interval.

The column values for the /ARBA/TVARV table are:

- Variable Name = /ARBA/GOODS_RECEIPTS_CREATE
- Selection Category = P
- INCL/EXCL = I
- Selection Value = <RFC Time Interval >

Exporting Amount-Based Receipts

The Ariba procurement solutions allows you to receive based on quantity or amount. For more information, see the Ariba Procurement User Guide.

For amount-based receipts, the Ariba integration code maps the amount to the quantity in SAP, and sets the price to 1. For example, if the amount of the receipt in the Ariba procurement solutions is $95, it is exported to SAP as an receipt with a quantity of 95 and with a price of $1.

Limitations for Integrating Receipts

In a receipt, if you have both accepted quantity and rejected quantity, only accepted quantity is integrated with the SAP system. All rejected quantities are ignored.
Integrating payments

Related Information

Integrating non-PO (FI) invoices [page 232]

About payment requests

After an invoice has been approved, the final step is to send a payment to the supplier. When it creates an invoice reconciliation document, the Ariba Procurement Solution also creates a new approvable document, called a payment request, to track the payment process.

Most payment requests represent amounts due to a supplier, such as for an unpaid invoice. The Ariba Procurement Solution also provides support for payment requests associated with credit memos. A credit memo is an invoice that represents a refund owed to your company by the supplier. For example, if your company purchased equipment that was later returned, the supplier might first issue an invoice and then (after the return) a credit memo, which represents the amount due to be refunded to you. The Ariba Procurement Solution creates credit payment requests to represent credit memos.

The Ariba Procurement Solution exports payment requests that use the External System payment model only. The Ariba Procurement Solution does not export partial payments, even if you import a partial remittance from your SAP system.

When exporting payments, discounts are supported both at header level and line level.

Note

- PO based and non-PO based invoices are integrated to MM module of SAP ERP. Ariba provided ABAP programs ensure that all MM invoices create an FI document (accounting document) internally.
- There is no default contract integration with SAP ERP. Therefore, contract reference information is not sent to SAP ERP along with non-PO invoice integration. To do this, customers need to perform customizations accordingly.

Prerequisites for integrating payments

Ensure the following prerequisites are met before proceeding with the payment export:

1. Tax Data is populated (includes Tax Code Pull, TaxCodelookup, TaxRatelookupbuTaxCode)
2. Remittance Location Information is loaded
3. Payment Term and Method are loaded
Integration events for exporting payment requests

The following table describes the integration event that exports payment requests.

<table>
<thead>
<tr>
<th>Integration Event</th>
<th>Integration Channel</th>
<th>Integration Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>PaymentExport</td>
<td>Ariba Web Services Channel</td>
<td>Exports payment request data to SAP ERP. The display name for this integration event in Ariba Administrator is Export Payment Requests.</td>
</tr>
</tbody>
</table>

You can run this integration event by using either Ariba Administrator or the Ariba Data Transfer Tool. For information on using the Ariba Data Transfer Tool, see the Ariba On-Demand Integration Toolkit Guide.

Limitations for integrating payments

Payments integration in this release has the following limitations:

- Invoices are sent to the SAP ERP after reconciliation in the Ariba Procurement Solution. Hence, these are reconciled invoices sent to SAP ERP.
- The Ariba Procurement Solution sends Tax Code to SAP ERP. During invoice reconciliation, if you accept any tax variances, it can fail in SAP ERP based on the ERP system set up.
- Taxes shipping and handling.
- Invoices with zero value amount for line items.
- Invoices with line items containing the net price and zero value amount. In such a case, the Ariba Procurement Solution creates an invoice only for the line items that have Net Price.

Integrating remittance

Importing remittance advice

Remittance advice is an electronic or paper document that identifies and provides details about the open invoices that are in the process of being paid. Remittance details include the payment methods, bank information, and a list of paid invoices with discount and paid amounts.

Remittance import is available through the Ariba Settlement module. You can import remittance advice for payments created for the following, within a specified time period:

- Invoices
- Credit and debit memos

Remittance import is also available for:

- Future-dated payments
- Voided payments, including void checks and reverse payments

To process remittances, maintain the following payment document type entry in the table /ARBA/TVARV before running the program /ARBA/CR_REMITTANCE_EXPORT:
Limitations for Integrating Remittance

SAP does not support remittance imports in the following scenarios:

- Payment document contains withholding tax.
- Residual payments are involved.
- Payment document has been reset.
- Payment document is an intercompany payment.
- Payment document is a down payment.

Process Flow for Integrating Remittance

1. Run or schedule the IncrementalRemittanceImport schedule task from Ariba procurement solutions. This incremental task uses the `defaultStartDate` specified in the `ScheduledTask.table` as input for the first time. The date and time sent by SAP is used as input for all subsequent imports.

   **Note**
   With the date and time sent from Ariba procurement solutions, SAP returns all payments completed in the SAP system (including Full and partial payments).

2. Each payment document in SAP is mapped to create a payment transaction in Ariba procurement solutions, or, in the case of a void document, is mapped to update a payment transaction in Ariba procurement solutions.

   In SAP, it is possible to cancel a payment document for a specified payment, or to re-issue a check for a payment document without cancelling the payment document.
   - If a payment document is cancelled in SAP, the corresponding payment transaction imported into Ariba procurement solutions is voided.
   - If a check is re-issued in SAP without cancelling the payment document, the entire payment transaction in Ariba procurement solutions is voided, and a new payment transaction with a new check number is created.

Remittance Advice Import Considerations

SAP does not support remittance import in the following scenarios:

- The payment document contains withholding tax.
- The payment document has been reset.
The payment document is an intercompany payment.

The payment is for an invoice created for an expense report.

The payment document is a down payment or a residual payment.
The following considerations pertain to remittance advice import:

- When you use the Check Payment method, remittance advice is imported only after the checks have been printed.
- If you make a payment to a special general ledger (G/L) account, remittance advice is not imported into Ariba procurement solutions. Common special GL accounts might include:
  - Down payments
  - Bills of exchange
  - Guarantees
  - Security deposits
  - Individual value adjustments
  - Amortization
  - Interest payable
- The payment documents generated in SAP from manual payment might not include the payment method by default as this is not a required field in SAP. However, in Ariba procurement solutions, the payment method is a required field and if it is empty in <RemittanceEventName>.response, the load fails.
- The payment documents generated in SAP from cancel payment might not include the payment method by default as this is not a required field in SAP. However, in Ariba procurement solutions, the payment method is a required field and if it is empty in <RemittanceEventName>.response, the load fails. To avoid a load failure, when mapping the values in Ariba procurement solutions map the payment method as "C" (Check) for cancel Payments, when there is no value set in SAP.
- For manual payment, if you print the check after /ARBA/CR_REMITTANCE_EXPORT runs, then /ARBA/ CR_REMITTANCE_EXPORT imports the check number into Ariba procurement solutions during the next remittance advice run and updates the Payment Number in the Payment Transaction object. However, it is not exported to the supplier on the Ariba Network.
  /ARBA/CR_REMITTANCE_EXPORT imports down payments as credit-line items after subsequent payments are made. Down payments are not imported at the time of creation.

For additional information on ERP considerations when importing remittance advice, see the Ariba Settlement Guide.

**Integrating real-time budget check**

This feature is supported on Ariba Procurement Solution integrated with SAP ERP versions starting with SAP ERP 6.0 up to SAP ERP 6.0 EHP 7.

Buyers using SAP ERP must complete the configuration tasks required for integration with the Ariba Procurement Solution to enable budget checks. The budget check functionality is available for WBS elements, Internal Orders, Funds Management, and Grants Management in SAP ERP.

In addition, using the ARBA/MASTER_DATA_EXPORT program, buyers can extract the Funds Management master data from SAP ERP to integrate Funds Management accounting with the Ariba Procurement Solution. Using the Full Load option, you can extract all the fields of the Funds Management master data, excepting Earmarked Funds. You can extract Earmarked Funds using the incremental load only. Funds Management derivation is supported on the following transactions:
Requisitions (Internal Orders, WBS, Funds Management, and Grants Management)
Change Requisitions (Internal Orders, WBS, Funds Management, and Grants Management)
Purchase Orders
Change Purchase Orders
Invoices

To enable real-time budget check, you must configure the integration scenario for the following:

1. Budget check
2. Requisition

For more information about configuring the integration scenario, see About Configuring the Integration Scenario [page 28].

To configure enhancements for budgets, see How to configure enhancements for budgets [page 224].

Prerequisites

- **Ariba Procurement Solution**

While setting up the end-points for integration with SAP ERP, you have the option to configure the appropriate BAPIs that determine the type of presubmission budget availability checks on requisitions when you click the Check button. You can choose one of the following function modes when you configure the end-points for integration with SAP ERP:

- **Full requisition check including budget availability checks**: This mode is available from Cloud Integration release 9.0, if you use either the Materials Management module or the Materials Management module used in combination with the Financial Accounting module of SAP ERP.

  When you configure the end-points for full requisition check, the Ariba Procurement Solution performs the following actions when you click the Check button on a requisition:

  - Sends a copy of the requisition to SAP ERP to validate the accounting information on the requisition
  - (If the Funds Management accounting functionality is enabled) Populates accounting information for the Funds Management account assignment fields
  - Performs budget checks in SAP ERP for availability of funds for the requisition

  To configure the full requisition check, use the following BAPIs:

  - BAPI_PR_CREATE (for requisition create)
  - BAPI_PR_CHANGE (for requisition change)

- **Budget availability check**: This mode is supported by the Financial Accounting module in SAP ERP.

  When you configure the end-points for budget availability check, the Ariba Procurement Solution performs the following actions when you click the Check button on the requisition to perform pre-submission budget checks:

  - (If the Funds Management accounting functionality is enabled) Populates accounting information for the Funds Management account assignment fields
  - Performs budget checks in SAP ERP for availability of funds for the requisition

  While configuring the end-points for budget availability check, use the BAPI BAPI_ACC_PURCHASE_REQUI_CHECK.

**Note**

We recommend that you choose full requisition check while configuring the end points for web services to ensure that requisitions are fully validated before users submit them for approvals.
Irrespective of the end-point configurations that you choose for the pre-submission budget checks, when you click the **Submit** button to submit a requisition for approval, SAP ERP validates the requisition for both accounting combinations and availability of funds.

### SAP ERP

- Before creating requisitions, if not already applied, apply the following SAP Notes in your SAP ERP:
  - SAP Note **1279915** - *Purchase requisition BAPI: Release date is not transferred.*
  - SAP Note **970686** - *Item category change deletes source of supply.*
  - SAP Note **962638** - *Valuation price is not output.*
  - SAP Note **1510700** - *BAPI_PR_CHANGE: account assignment failure.*

- Set the following variables in the table **/ARBA/TVARV** in your SAP ERP to support real-time budget checks on requisitions:
  - **/ARBA/BAPI_PR_CRT_ALLOW_DUPS**
  - **/ARBA/PURREQ_DOC_TYP**
  - **/ARBA/HELD_PR_NO**

- To configure the integration scenario when using the **Full requisition check including budget availability checks** mode, use the following communication channels:

<table>
<thead>
<tr>
<th>Function</th>
<th>Communication Channel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requisition submit</td>
<td>CCT_ReqSubmit_Sender_CIX</td>
</tr>
<tr>
<td>Requisition budget check</td>
<td></td>
</tr>
<tr>
<td>Requisition revert to earlier version</td>
<td></td>
</tr>
<tr>
<td>Requisition withdraw</td>
<td>CCT_ReqWithdraw_Sender_CIX</td>
</tr>
</tbody>
</table>

**Note**

X in the communication channel name denotes the Cloud Integration release.

- To display an error message in the Ariba Procurement Solution when you exceed the budget, perform the following in SAP ERP:
  - Set appropriate tolerance limits for the **General Budget Profile** and the **Budget Profile Investment** using the **Define Tolerance Limits for Availability Control**.
  - Set the parameter **Availability control action** to 3 (**Error message**).

**Related Information**

- How to edit **/ARBA/TVARV** parameters [page 49]
- Configuring the Integration Scenario [page 31]
Limitations

- The Ariba Procurement Solution integrated with SAP ERP does not support both web services and file channel to integrate all transactional documents containing the budget information. Ariba recommends that you use only web services to integrate corresponding transactional documents associated with a requisition containing the budget information.

- When users edit a requisition in the Ariba Procurement Solution and change a multiple split-accounting to a single account assignment, the **Distribution** flag indicator for the requisition cannot be reset to zero on SAP ERP.
  The Ariba Procurement Solution integrated with SAP ERP resets the **Distribution** flag indicator to 1 (Distribution by Quantity) and sets the **Partial Invoice** indicator to 2. Ariba recommends that you update the requisition only on the Ariba Procurement Solution. However, if you edit the requisition in SAP ERP and change the multiple split accounting to a single account assignment, the following error appears: If only one account assignment item, choose "Single Account Assignment.

- The Ariba Procurement Solution integrated with SAP ERP does not support non-PO based invoices with earmarked fund as the signature of the BAPI `BAPI_INCOMINGINVOICE_CREATE` does not have the following fields for the earmarked fund:
  ○ Document Number for Earmarked Funds
  ○ Earmarked Funds Document Item
  As a result, for non-PO based invoices, the amount is consumed from the overall budget instead of the earmarked fund.

How to configure enhancements for budgets

**Context**

To configure the implicit enhancement for budgets, perform the following steps:

**Procedure**

1. In the SAP ERP main screen, navigate to `SAP Menu > Tools > ABAP Workbench > Development > ABAP Editor`.
   The **ABAP Editor: Initial Screen** screen appears.
2. Open `include LACC4P10` in the display mode.
3. Click `Menu > Program > Enhance`.
4. Click `Menu > Edit > Enhancement Option > Show Implicit Enhancement options`.
5. To add the enhancement, click on the line containing quotes ('"') after the code as shown:
6. Click **Menu > Edit > Enhancement Option > Create Implementation**.
7. Click **Declaration**.
8. Click **Create** in the next screen.
9. Enter the name for the enhancement implementation. For example, `ZPREQ`.
10. Enter a short description.
11. Click the green colored check mark.
12. Enter the package name and transport number in the next screen.
13. Choose the newly created enhancement and click the green checkmark.
14. Enter the following code:

```
DATA: L_CALL_TYPE TYPE FLAG.
CLEAR L_CALL_TYPE.
CALL METHOD /ARBA/CL_PURREQ_P2P=>BUD_CHK_DOCTYPE_CHECK
  IMPORTING
    ARIBA_CALL = L_CALL_TYPE.
  IF L_CALL_TYPE = 'X'.
    EXIT.
  ENDIF.
```
15. Activate the program.

### Integrating funds management

When funds management accounting functionality is enabled, funds management master data from the SAP ERP system is loaded in the Ariba Procurement Solution. This functionality allows the Ariba Procurement Solution to derive accounting values from the SAP ERP for funds management account assignment fields on requisitions and invoice reconciliation documents based on other accounting information that is included in such documents.

### Fields available for funds management integration

The following screens in SAP ERP include the checkbox **Enable Funds Management** under **Funds Management Data**:

- **Export All Master Data Required For On-Demand Application**
- **Pull All The Master Data In Diff Lang Required For Ariba Application**

Check **Enable Funds Management** to display the following checkboxes:
Integrating funds management master data

To integrate funds management master data with the Ariba Procurement Solution, see

- Integrating master data using the user interface [page 142]
- About importing data in multiple languages [page 149]

To configure funds derivation integration scenario, see About Configuring the Integration Scenario [page 28].

Maintaining filtering for funds management extract

To restrict the data that the Ariba Procurement Solution RFC imports, make a corresponding entry in the /ARBA/TVARV table by creating a variable Variable Name.

Note

Prefilter name is the same as the class name for funds management extract but without the CL_prefix. For example:

- Class name: /ARBA/CL_FUND_EXPORT
- Prefilter name: /ARBA/FUND_EXPORT

For more information about filtering data, see Static filtering [page 133].

To customize the funds management extract from SAP ERP to the Ariba Procurement Solution, see Customizing Funds Management extracts [page 150].

Related Information

Prerequisites [page 222]
Limitations [page 224]
Integrating taxes and charges

This feature is supported on Ariba Procurement Solution integrated with SAP ERP versions starting with SAP ERP 6.0 up to SAP ERP 6.0 EHP 7.

The following transactions have been enhanced to support this feature:

Table 9: Supported Transactions

<table>
<thead>
<tr>
<th>Using Web Services</th>
<th>Using File Channel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requisitions (with budget check functionality)</td>
<td>-</td>
</tr>
<tr>
<td>Change Requisitions</td>
<td>-</td>
</tr>
<tr>
<td>Purchase Orders</td>
<td>Purchase Orders</td>
</tr>
<tr>
<td>Change Purchase Orders</td>
<td>Change Purchase Orders</td>
</tr>
<tr>
<td>Invoices</td>
<td>Invoices</td>
</tr>
</tbody>
</table>

Note

The Ariba Procurement Solution sends discount and charges to SAP ERP only in amount and not in percentage values.

To configure taxes and charges on requisitions, purchase orders, and invoices, you can use the following configuration parameters available on the SAP Easy Access Transaction Code for Ondemand Procurement screen under SAP Menu Transactional Data Configuration Parameter:

- Define tax check at company code level
- Maintain Ariba condition mappings in SAP
- Maintain Withholding Tax Type for Buyer tax type

The following sections describe the behavior in SAP ERP for requisitions, purchase orders, and invoices with taxes, charges, and discounts received from the Ariba Procurement Solution:

Requisitions

Note

The budget reserved amount is equal to the final requisition amount derived in SAP ERP after calculating all applicable taxes, charges, and discounts.

The following examples describe the behavior of SAP ERP when requisitions received from Ariba Procurement Solution contain a:

- line item with a charge From Ariba Procurement Solution: A line item of USD 100 has a shipping charge of USD 10.
Result in SAP ERP: Single line with amount = USD 110.

**line item with charges**

From Ariba Procurement Solution: A line item of USD 100 has a shipping charge of USD 10 and a handling charge of USD 5.

Result in SAP ERP: Single line with amount = USD 115.

**line item with a charge and tax on charge**

From Ariba Procurement Solution: A line item of USD 100 has a shipping charge of USD 10 along with a tax amount of USD 5.

Result in SAP ERP: Two lines:
- First line contains the actual line item amount = USD 100.
- Second line contains the total amount of shipping charge and tax on the shipping charge = USD 15.

**line item with a non-deductible tax amount**

From Ariba Procurement Solution: A line item of USD 100 has a non-deductible tax of USD 5.

Result in SAP ERP: Single line with amount = USD 105.

**line item with a deductible tax amount**

From Ariba Procurement Solution: A line item of USD 100 has a deductible tax of USD 5.

Result in SAP ERP: Single line with amount = USD 100.

**line item with a discount**

From Ariba Procurement Solution: A line item of USD 100 has a discount of USD 10.

Result in SAP ERP: Single line with amount = USD 90.

---

**Purchase Orders**

The following examples describe the behavior of SAP ERP when purchase orders received from Ariba Procurement Solution contain a:

**line item with a charge**

From Ariba Procurement Solution: A line item of USD 100 has a shipping charge of USD 10.

Result in SAP ERP: Single line with amount = USD 110.

*Shipping charge* = USD 10 is maintained as a condition in the **Conditions** tab of the purchase order transaction.

**line item with charges**

From Ariba Procurement Solution: A line item of USD 100 has a shipping charge of USD 10 and a handling charge of USD 5.

Result in SAP ERP: Single line with amount = USD 115.

*Shipping charge* = USD 10 and *Handling charge* = USD 5 are maintained as separate conditions in the **Conditions** tab of the purchase order transaction.

**line item with a charge and tax on charge**

From Ariba Procurement Solution: A line item of USD 100 has a shipping charge of USD 10 along with a tax code, for example, I1.

Result in SAP ERP: Two lines:
- First line contains the actual line item amount = USD 100.
- Second line contains the shipping charge (net price) and tax on the shipping charge. *Tax code* I1 is maintained in the **Invoice** tab of the purchase order transaction.
From Ariba Procurement Solution: A line item of USD 100 has a tax code, for example, I1.

Result in SAP ERP: Single line with amount = USD 100.

**Tax code** I1 is maintained in the **Invoice** tab of the purchase order transaction.

From Ariba Procurement Solution: A line item of USD 100 has a discount of USD 10.

Result in SAP ERP: Single line with amount = USD 90.

Discount = USD 10 is maintained as a condition in the **Conditions** tab of the purchase order transaction.

### Invoices

The following example describes the behavior of SAP ERP when invoices received from Ariba Procurement Solution contains a:

**Header with a withholding tax**

From Ariba Procurement Solution: Total amount = USD 100 has a header with a withholding tax WT1 = USD 5.

Result in SAP ERP: Withholding tax = USD 5 along with the **Tax Code** and **Tax Type** is displayed in the **Withholding Tax** tab of the invoice transaction.

### Related Information

**Tables for taxes and charges** [page 53]

### Prerequisites

- You must maintain the following tables in SAP ERP for:
  - **Requisitions**: `/ARBA/TAX_CHECK` - To specify the company code for which the non-deductible tax amount (tax on item or tax on charges) is considered while reserving the budget for requisitions under a particular company code.
  - **Purchase orders**: `/ARBA/COND_MAP` - To specify the condition types in the Ariba Procurement Solution to the relevant condition types in SAP ERP.
  - **Invoices**: `/ARBA/WHT_TYPE` - To specify the withholding tax type in the Ariba Procurement Solution to the relevant withholding tax code in SAP ERP.

To support withholding tax on invoices, you must configure and enable the withholding tax at the vendor level in SAP ERP. The tax code configuration must be in sync between the Ariba Procurement Solution and SAP ERP.

- For purchase orders and change purchase orders, you must maintain the `/ARBA/ENABLE_TLC` parameter in the `/ARBA/TVARV` table in SAP ERP. The `/ARBA/ENABLE_TLC` parameter is required to update the SAPPOLineNumber in the Ariba Procurement Solution.

For more information about setting variables in the `/ARBA/TVARV`, see **How to edit /ARBA/TVARV parameters** [page 49].
To avoid calculation errors in SAP ERP, you must not exceed the configured decimal limit when entering values in the Ariba Procurement Solution.

**Related Information**

Tables for taxes and charges [page 53]

**Limitations**

- Conditions must be amount based (fixed value). Percentage is not supported.
- When a new line item contains tax on charges on a change purchase order, addition of split on the line item is supported only from SAP ERP 6.0 EHP5 and higher.
- Withholding tax is not supported for the following:
  - line items on invoices
  - remittances
- Header-level credit memos and line-level credit memos with tax code are not supported.
- When real-time budget check is enabled in the Ariba Procurement Solution, SAP ERP does not support changing units of measure in a purchase order referencing a requisition. If you attempt to change the units of measure when creating the purchase order in the Ariba Procurement Solution, the following error message appears in SAP ERP:
  "Order unit XX not convertible into unit XX of requisition"

**Integrating advance payments**

This feature enables buyers to make advance payments for a purchase order to a supplier. Suppliers can use these advance payments to procure raw material or hire resources to process the orders. Buyers can adjust the advance payments against the invoices received for a purchase order.

**Note**

This feature requires minimum SAP ERP 6.0 SPS02 (SAP_APPL 600 SPS02 and SAP_BASIS 700 SP29) version.

In SAP ERP, the advance payment against an advance payment request is made using the F48 transaction through manual posting or automatically through the payment run, depending on system configurations.

Ariba Procure-to-Pay exports fully approved advance payment request documents to the SAP ERP to process the payments. The SAP ERP sends a confirmation to Ariba Procure-to-Pay when the advance payment is successfully processed. The advance payment amount can be adjusted in the invoice during invoice reconciliation.
Prerequisites

SAP ERP

- Implement SAP Note 2274732 - Incorrect document status when posting noted items with BAPI_ACC_DOCUMENT_POST.
- Implement BAdI definition BADI_ACC_DOCUMENT.
  - Add the following code to the method:
    ```java
    INCLUDE /arba/advpymnt_badi_include IF FOUND
    ```
  - Add a filter in the reference transaction field BKPF.
  
  For more information about implementing a BAdI, see How to Implement a BAdI.

- Maintain the following parameters in the table /ARBA/TVARV:
  - /ARBA/EVENT_APR - to get data for advance payment requests from Ariba Procure-to-Pay to SAP ERP.
  - /ARBA/APR_PAY_REMIT_EXPORT - to support advance payment remittance using direct connectivity.

- Maintain a temporary directory to download CSV files while running the advance payment using direct connectivity. Specify the following variable in the table /ARBA/TVARV:
  - /ARBA/APR_CREATE - for advance payment requests creation.
  - /ARBA/APR_CANCEL - for advance payment requests cancellation.

- Run the Business Transaction Event (BTE) FIBF and then perform the following steps to maintain the parameters for the advance payment remittance:
  - Settings ➤ Process Modules ➤ of a customer ➤
    - Process Code: 00005010
    - Function Module: /ARBA/BTE_ADV_PAY_REMITTANCE
    - Product: /ARBA/AP or as per your requirement.

  - Settings ➤ Product ➤ of a customer ➤
    - Product: /ARBA/AP or as per your requirement.
    - Text: Ariba advance payment
    - Choose Active Flag.

Note

If you support both, Ariba Network Adapter for SAP NetWeaver and Ariba Procurement Solution integrated with SAP ERP, then you must:
- Replace /ARBA/BTE_ADV_PAY_REMITTANCE with /ARBA/BTE_00005010 in the Function Module.
- Ensure that the supported products are active in the Customer Products.
Related Information

Prerequisites for exporting payments [page 189]
Advance payment parameters [page 55]
How to maintain a temporary directory while running transaction data directly [page 70]

Limitations

SAP ERP

- Non-purchase order based advance payments are not supported.
- Advance payment requests are supported from Ariba Procure-to-Pay to SAP ERP but not from SAP ERP to Ariba Procure-to-Pay.
- SAP ERP supports advance payments only at the line level although Ariba Procure-to-Pay supports advance payments at both the header level and line level.
- The clearing document created as a part of invoice adjustment is not shown on the Follow-On Documents tab. This clearing document can be seen under the Purchase Order History tab.

Ariba Procure-to-Pay

- An advance payment referencing multiple purchase orders is not supported.
- Only amount-based advance payment requests are supported.
- Advance payment requests or advance payment remittances are not supported for line items with split accounting.

Integrating non-PO (FI) invoices

This feature enables buyers using Ariba Procurement Solution integrated with SAP ERP to create non-PO (FI) invoices in SAP ERP.

The following graphic shows the process flow for creation of non-PO (FI) invoices:
Creation of non-PO (FI) invoice in SAP ERP supports the current non-PO (MIRO) invoice functionalities such as discounts and withholding taxes. In addition, it has the following advantages when compared to the logistics invoice verification API BAPI_INCOMINGINVOICE_CREATE to create non-PO (MIRO) invoices:

- Does not require MM module on SAP ERP to integrate payments from Ariba Procure-to-Pay.
- Supports asset posting.

To support non-PO (FI) invoices in SAP ERP:

- RFC /ARBA/BAPI_INVOICE_CREATE has been enhanced.
- BAPI BAPI_ACC_INVOICE_RECEIPT_POST is called in the RFC.
- BAdI /ARBA/INVOICE_CREATE has been enhanced and uses the method FI_INVOICE_CREATE.

**Note**

Shipping and special handling as unplanned delivery costs are posted as G/L lines.

**Related Information**

- Non-PO (FI) invoices [page 62]
- Configuring the Integration Scenario [page 31]
- Exporting invoices [page 188]
- /ARBA/INVOICE_INTEGRATION Package [page 307]

**Prerequisites**

**SAP ERP**

- Maintain the following parameters in the table /ARBA/TVARV:
  - /ARBA/USE_FI_POSTING_NONPO to use FI based posting for non-PO invoices.
  - /ARBA/NOPO_INVOICE_FI_DOC_TYPE to set the document for an FI invoice. By default the document type is KR.
  - /ARBA/ASSET_TR_TYP_FI_NONPO to maintain the asset transaction type, if you use asset accounting assignment. By default the asset transaction type is 020.

- If you use the withholding tax or asset account assignment functionality, then you must:
  - Maintain the table /ARBA/WHT_TYPE to specify the relevant withholding tax code in SAP ERP corresponding to the tax type in the Ariba Procurement Solution.
  - To support withholding tax on invoices, you must configure and enable the withholding tax at the vendor level in SAP ERP. The tax code configuration must be in sync between the Ariba Procurement Solution and SAP ERP.
    | 1. Create a custom project by running the transaction code CMOD and link it to the enhancement ACBAPI01.
    | 2. Use the function exit EXIT_SAPLACC4_001 and create the include ZXACCU15.
3. Place the SAP Ariba provided include program `/ARBA/FILL_MISS_FI_NONPO` in the `ZXACCU15` as follows:

```plaintext
INCLUDE /ARBA/FILL_MISS_FI_NONPO IF FOUND
```

The earlier steps are in accordance with the following SAP Notes:

- SAP Note 556311 - BAPI in the extended withholding tax (for withholding tax)
- SAP Note 487722 - Using EXTENSION1 for accounting BAPIs (for withholding tax and asset account assignment)

## Limitations

### SAP ERP

- Tax-relevant posting in a company code in which the line-by-line tax calculation is active is not supported. For more information, see section 4.6 Postings with a customer program in the SAP Note 2011553 - FAQ: Error FF 817 or FF 818 with postings via BAPI_ACC_DOCUMENT_POST.
- Although tax code with dual tax rates is supported, both the tax rates must be either with a G/L account or with a non-GL account. Dual tax rates with both G/L account and non-GL account together are not supported.
- Header-level credit memos and line-level credit memos with tax code are not supported.
- The withholding tax is supported only at the header level.

### Ariba Procure-to-Pay

Ariba Procure-to-Pay does not provide G/L account for asset lines. Therefore, the G/L account for asset lines is derived from the table `T095_ACI` based on chart of accounts.

If the table `T095_ACI` also does not include G/L account for asset lines, then the G/L account for asset lines is derived from the table `T095` based on the chart of accounts, account determination, and real depreciation area. The real depreciation area is always 01.

## Integrating service procurement

### About integration of service procurement in the Ariba Procurement Solution integrated with SAP ERP [page 234]

Prerequisites [page 236]

Limitations [page 237]

### About integration of service procurement in the Ariba Procurement Solution integrated with SAP ERP

Service procurement is a set of features that enables buyers to place orders for services, involves suppliers who fulfill the orders, and allows buyers to track, manage, and reconcile invoices for those orders.
This feature, available in the Ariba Procurement Solution integrated with SAP ERP, is applicable to the following solutions:

- Ariba Procure-to-Pay
- Ariba Procure-to-Order

The approvable documents associated with the service procurement process that flows across the Ariba Procurement Solution, SAP ERP, and Ariba Network are as follows:

- **Service requisitions**
  Buyers create service requisitions in the Ariba Procurement Solution to request for services performed by supplier organizations. While creating service requisitions, buyers have the option to validate the requisitions for availability of budgets in SAP ERP. If budget is available in SAP ERP and the response is successful, the Ariba Procurement Solution sends a request to SAP ERP to create a service requisition to reserve the fund. If the service requisition is deleted or canceled by the buyer in the Ariba Procurement Solution, the funds reserved for the requisition is released to the budget in SAP ERP.

- **Service orders**
  When a service requisition is fully approved, the Ariba Procurement Solution creates a service order corresponding to the requisition and sends the service order to SAP ERP. SAP ERP assigns an unique number to the service order and sends it to the Ariba Procurement Solution, which in turn sends the order to the supplier on the Ariba Network.
  The mandatory **Max Amount** field for service orders in the Ariba Procurement Solution is mapped to the **Overall Limit** under the tab **Limits** of the service orders in SAP ERP. Therefore, all service orders in SAP ERP include the **Limits** value.
  The **Expected Amount** field for service orders in the Ariba Procurement Solution is mapped to the **Expected value** under the tab **Limits** of the service order in SAP ERP. The **Expected Amount** is added to the **Net Price** condition for the main line item.

- **Service sheets**
  Suppliers performing a service create service sheets in Ariba Network in order to describe that service. Service sheets can contain both service and material goods lines. Suppliers can create a service sheet for one or more service lines on an associated service order and send it to the Ariba Procurement Solution.
  Buyers on the Ariba Procurement Solution can receive:
  - Service sheets with or without hierarchy. Only two level deep (parent and child) hierarchy is supported.
  - Service sheets containing planned and unplanned service lines
  - Service sheets with attachments
  - Service sheets with accounting splits on single and multiple service lines—both on parent line (split in the **Limits** tab) and child line
  The Ariba Procurement Solution, after approvals, sends the service sheet to SAP ERP.
  SAP ERP sends:
  1. Either a service sheet number to the Ariba Procurement Solution when the service sheet is successfully created or an error response if there is a failure.
  2. A response (approved, rejected, or processing) to the Ariba Procurement Solution and Ariba Network to update the service sheet status, after the buyer accepts, rejects, or revokes the service sheet after approval in SAP ERP.

- **Service invoices**
  Suppliers on Ariba Network create service invoices for the approved service sheets and send them to the Ariba Procurement Solution for reconciliation and approval. After approval, the service invoices are sent to SAP ERP.
Additional References

For more information about the service procurement documents, see the following documents on Ariba Connect:

- Purchasing Guide for Procurement Professionals
- Creating and Managing Invoices

Related Information

Configuring the Integration Scenario [page 31]
Configuring service sheets [page 74]
Troubleshooting integration of service procurement in the Ariba Procurement Solution integrated with SAP ERP [page 266]

Prerequisites

Ariba Procurement Solution

Enable the parameters required for service procurement. For more information, see the Customer Site Parameters guide.

SAP ERP

- **Service requisitions:**
  - Ensure that the BUSINESS FUNCTION:LOG MM_CI2 switch MM_SFWS_SC2 is active.
  - Review and implement the following SAP Notes as needed:
    - 1723449 - BAPI_PR_CHANGE: error message SE 721
    - 1914440 - BAPI_PR_CREATE/BAPI_PR_CHANGE: account assignment errors
    - 1947235 - BAPI: Decimal not considered for the creation of purchase requisition
    - 1978487 - BAPI: SE 181 issued during purchase requisition change using BAPI_PR_CHANGE
    - 2003042 - BAPI: SE601 issued during purchase requisition change
    - 2010358 - SPPR: Decimal value of the currency not considered for the display on limit
    - 2079845 - BAPI_PR_CREATE: G/L Account is not determined
    - 2195857 - SPPR: Multiple issues during the purchase requisition create/change
    - 2301884 - Issues With BAPI_PR_CHANGE
    - 2379204 - BAPI_PR_CHANGE fails with an error message SE601

- **Service orders:** Ensure that the purchase order scenario is integrated successfully.
Limitations

SAP ERP

- The following service order fields from the Ariba Procurement Solution are not mapped in SAP ERP:
  - Service Start Date
  - Service End Date
- More than two levels in the service hierarchy are not supported.
- Taxes and charges are supported only at the parent line level and not at the service child line level.
- Irrespective of the unit of measure (UOM) entered in the main line item for service requisitions in the Ariba Procurement Solution, quantity changes to 1 activity unit (AU) in service orders in SAP ERP. Service child line items in the service orders can have different quantity and UOM as needed.
- You cannot create service invoices with reference to service orders. You can create invoices only with reference to service sheets.

Integrating attachments support

About attachments with purchase orders and invoices in the Ariba Procurement Solution integrated with SAP ERP [page 237]

Prerequisites [page 238]

Limitations [page 239]

Viewing purchase order and invoice attachments in SAP ERP [page 239]

About attachments with purchase orders and invoices in the Ariba Procurement Solution integrated with SAP ERP

This feature, applicable to the Ariba Procurement Solution integrated with SAP ERP, supports file attachments with both purchase orders and invoices. File attachments are useful when:

- Buyers need to attach supporting memos, drawings, or faxes with their purchase orders
- Suppliers need to attach supporting documentation with their invoices

Using this feature, buyers can use attachments and comments with purchase orders and attachments with invoices in SAP ERP integrated documents. The attachments and comments are now sent to SAP ERP as part of the purchase order and invoice integration messages. In SAP ERP, they are processed as follows:

- **Attachments**
  - Header level using the SAP Generic Object Services (GOS) for purchase orders and invoices
  - Line level using the SAP Document Management System (DMS) for purchase orders

For purchase orders, along with the attachments on the header and line level, the following additional information is sent from the Ariba Procurement Solution to SAP ERP:

- File name: Specifies file name. If this field is left blank in the Ariba Procurement Solution, the file displays as unknown in SAP ERP.
○ File type: Specifies file extension.
○ File size
○ Comments on header and line level.
○ External comments flag: Indicates if comments are visible to suppliers.

If files fail to attach with purchase orders or invoices in the Ariba Procurement Solution, SAP ERP sends an email to the user. More than one user can receive error emails based on the number of email IDs maintained in the table /ARBA/TVARV. For information about the parameter to be maintained for purchase orders and invoices in the table /ARBA/TVARV, see Prerequisites [page 238].

![Note]

SAP ERP truncates file name of attachments exceeding 50 characters (inclusive of extension).

- **Comments**
  Comments are supported for purchase orders at both the header and the line level. The comments field allows you to enter free form text that can be associated with a purchase order. You use the comments field to enter any additional information about a purchase order.

**Related Information**

Configuring the Integration Scenario [page 31]
Summary of Ariba SAP packages [page 281]
How to enable support for attachments in the Integration Scenario [page 39]

**Prerequisites**

- **Ariba Procurement Solution**
  ○ **Purchase orders and invoices**
    To control the file formats that can be attached to purchase orders and invoices in the Ariba Procurement Solution use the parameter Application.Approvable.AllowedAttachmentExtensions. File formats that are not specified cannot be used as attachments. This parameter is useful for restricting file formats (such as EXE files) that might violate company security policies.
  ○ **Purchase orders and change purchase orders**
    Enable parameter Application.Ordering.AllowOrderAttachmentToERP.
  ○ **Invoices**
    Enable parameter Application.Procure.SendAttachmentsToERP.

  For more information, see the Customer Site Parameters guide.

- **SAP ERP**
  ○ **Purchase orders and invoices**
    Enable support for attachments in the SAP Process Integration.
  ○ **Purchase orders and change purchase orders**
    Maintain the following parameters in the table /ARBA/TVARV:
    ○ /ARBA/HEADER_TEXT_ID: To process header level comments. For example, F01.
- /ARBA/ITEM_TEXT_ID: To process line level comments. For example, F02.
- /ARBA/DMS_STORAGE_CATEGORY: To maintain DMS storage category. For example, DRW.
- /ARBA/PO_ATT_ERR_MAIL: To specify the email addresses for error notification emails.
  - Invoices
    - Maintain the following parameter in the table /ARBA/TVARV:
      /ARBA/INV_ATT_ERR_EMAIL: To specify the email addresses for error notification emails.
    - Implement SAP Note 1476350-GOS: Attachment list does not display all linked objects to view attachments for FI invoices.

Related Information

- DMS attachments storage [page 59]
- Purchase order comments [page 63]
- Attachments failure email-purchase orders and change purchase orders [page 57]
- Attachments failure email-invoices [page 56]

Limitations

- Attachments at the line level are not supported for invoices.
- Comments are not supported for invoices.
- Attachments for service line items in purchase orders are supported only at the parent line level and not at the child line level.
- The attachment size cannot exceed 100 MB either at the header or line level (purchase orders). If you attempt to exceed the file attachment size of 100 MB, SAP ERP and the Ariba Procurement Solution either prevent or reject the document with an error message.
- Multiple attachments are permitted, however:
  - The limit for combined attachment size is 100 MB.
  - (Only for purchase orders) The limit for the number of attachments per line item is 20.

Viewing purchase order and invoice attachments in SAP ERP
How to view header level attachments with purchase orders

Context

To view purchase order attachments at the header level in SAP ERP, perform the following steps:

Procedure

1. Open a purchase order with attachments received from the Ariba Procurement Solution.
2. Click the dropdown (upper left) Services for Object Attachment list. If no attachments are available, the option Attachment list is grayed.
   A new windows appears and lists the available attachments.
3. Double-click the required attachment to view it.

How to view line level attachments with purchase orders

Context

To view purchase order attachments at the line level in SAP ERP, perform the following steps:

Procedure

1. Open a purchase order with attachments received from the Ariba Procurement Solution.
2. Choose the required line in the purchase order from Item Overview and click the icon Documents.
   The window Purchase Order Item - Link to Documents appears.
3. Double-click the attachment.
   The window Display Document appears.
4. Under Document Data Originals, click the attachment.
   The popup SAP GUI Security appears.
5. Click Allow to grant security access
   The attachment opens with the associated program.
How to view header level comments with purchase orders

Context

To view purchase order comments at the header level in SAP ERP, perform the following steps:

Procedure

1. Open a purchase order with attachments received from the Ariba Procurement Solution.
2. Navigate to the header text object according to your configuration and display the text: For example, F01 or F02:
   - F01: Header > Texts > Header text
   - F02: Header > Texts > Header note

How to view line level comments with purchase orders

Context

To view purchase order comments at the line level in SAP ERP, perform the following steps:

Procedure

1. Open a purchase order with attachments received from the Ariba Procurement Solution.
2. Navigate to Item > Texts > Item text to view the comments.
How to view header level attachments with invoices

Context

To view invoice attachments at the header level in SAP ERP, perform the following steps:

Procedure

1. Open an invoice with attachments received from Ariba Procure-to-Pay.
2. Click the dropdown (upper left) | Services for Object > Attachment list |. If no attachments are available, the option Attachment list is grayed.
   A new windows appears and lists the available attachments.
3. Double-click the required attachment to view it.

How to customize an existing integration event

Prerequisites

Before proceeding with the customization, ensure the following prerequisites are met:

- The required fields have been added using the Customization Manager or any other customizing tool.
- The SAP ERP code (Custom RFC among others) has been modified to handle the newly introduced fields.

Procedure

1. Generate the required WSDL.
2. Reload the WSDL file in the External Definition, which can be found at the following location in the SAP XI Integration Repository.
   SWC > NameSpace > Interface Objects > External Definitions
3. Save the file and activate it.
4. Import the RFC into Imported Objects on NetWeaver XI/PI. Save and activate it.
5. Now, re-import the modified RFC from the SAP ERP system. Save and activate it.
6. Open the appropriate Messaging Mapping object in edit mode and load the new WSDL schema.
7. Complete the mapping between the new elements in the source and target structures.
8. Save and activate.
Integrating transactional data directly using the user interface

About integrating transactional data directly using the user interface [page 243]
Configuring transactional data directly using the user interface [page 244]

About integrating transactional data directly using the user interface

Buyers using the Ariba Procurement Solution can integrate transactional data directly from the Ariba Procurement Solution to SAP ERP using the Direct Connectivity option.

ℹ️ Note

This feature is supported on SAP ERP 6.0 SPS02 (SAP_APPL 600 SPS02 and SAP_BASIS 700 SP18) and higher versions.

In direct connectivity, SAP ERP sends a request to the Ariba Procurement Solution. The request in its header contains all the required parameters for successful extraction of transactional data. Parameters include event name, operation, and time stamp. The download of files to SAP ERP involves two synchronous proxies with a \( \text{Request} \rightarrow \text{Response} \rightarrow \text{Request} \rightarrow \text{Response} \) approach.

Prerequisites

You must download and import the transport requests into your SAP ERP system.

Ensure that you do the following:

- Create a logical port in the SOAMANAGER for the consumer proxy class: \( /\text{ARBA/CO_MIOUT_SYNC_ARIBA_DOWN}. \)
- Implement the following SAP Notes:
  - 1402826 - Supplement to Note 548131
  - 1716777 - Runtime error IMPORT_WRONG_END_POS when displaying class
Before you export the transactional data, you must create the Authorization Object class.

Maintain the parameter /ARBA/TEMP_DIRECTORY in the table /ARBA/TVARV before you export the transactional data for the following transactions:
- Purchase Order Header Status
- Remittances
- Advance Payments

For more information see, How to maintain a temporary directory while running transaction data directly [page 70].

Create a logical file path for the ISMW_ROOT. This is only required to export data for purchase order header status, remittances, and any status updates sent for the transactions through the Direct Connectivity Integration method. For more information, see How to create a logical file path [page 158].

Limitations

- Administrators do not get email notifications when an error occurs during the transactional data integration.
- Administrators can only view the error messages in the runtime monitor of the SAP ERP and SAP Process Integration.
- The error messages do not store a record of all the error conditions in the transaction SLGI.

Configuring transactional data directly using the user interface

Perform the following procedures to configure transactional data directly using the user interface:

1. How to install client certificates [page 244]
2. Defining authorizations for running Ariba Procure-to-Pay transactions [page 90]
3. How to define site name in the maintenance table [page 245]
4. How to configure the SOAMANAGER for ECC 6.0 EHP5 and ECC 6.0 EHP6 systems [page 246]
5. How to configure client certificates to import transactional data directly [page 247]
6. How to run the transactional data integration tasks [page 249]

How to install client certificates

Context

To ensure that the connection request is originating from a known client, you must install client certificates before importing transactional data using direct or mediated connectivity.
Procedure

1. In SAP ERP, run the transaction code STRUST and click SSL Client SSL Client(Anonymous).
2. Click the button Import Certificates and then choose the path where you exported the certificates.
3. Click Allow and then click Continue to download the certificates.
4. Click Add to Certificate List.

You see a message that you have added the certificates successfully.

**Note**
Ensure that you save the newly added certificates. This notifies the ICM about the new certificates you have just imported.

5. After you have saved the entries in the STRUST, run the transaction code SMICM.
6. On the menu Administration, click ICM Exit Soft Global.
7. On the page ICM Monitor, restart the services.
8. On the menu Administration, click ICM Exit Hard Global. Restart the services for the Exit Global.

The certificate is now displayed in STRUST.

How to define site name in the maintenance table

**Context**

You can define the direct connectivity parameters by running the transaction code SM30 in the table /ARBA/AUTH_PARAM. Defining site name is required for mediated connectivity using SAP Process Integration and direct connectivity.

**Procedure**

1. Run the transaction code SM30.
2. Enter /ARBA/AUTH_PARAM.
3. Click Maintain.
4. Choose ARIBA from the dropdown under Solution.
5. Enter the site name under Realm.
6. **Optional:** Specify the wait duration between the retry attempts in Wait (in minutes).
How to configure the SOAMANAGER for ECC 6.0 EHP5 and ECC 6.0 EHP6 systems

Context

To establish connectivity for transactional data between the SAP ERP and the Ariba Procurement Solution, you must define the endpoints for the service by configuring SOAMANAGER. You must create two logical ports for transactional data:

- Download port: to download the transactional data from the Ariba Procurement Solution
- Upload port: to upload the status to the Ariba Procurement Solution

To create the logical ports, you must perform the following procedure separately, for download and upload:

Procedure

1. Run the transaction code SOAMANAGER.
2. In the Service Administration tab:
   - For EHP5, click Single Service Configuration
   - For EHP6, click Web Service Configuration
3. Select the following in the Search tab:
   - In the pulldown Search By, choose Consumer Proxy.
   - In the pulldown Search Pattern, enter MIO*.
   - (only for EHP5) In the pulldown Field, choose Both Names.
   The following Proxy Class appears:
   - For download: MIOut_Sync_AribaDownload.
   - For upload: MIOut_Sync_AribaUpload.
4. Click Apply Selection.
5. Click Create Logical Port.
6. In the SOA Management page, enter the port name in the Logical Port field.
7. In the field Description, enter a description.
8. Check Logical Port is Default.
9. In the Configuration Type, choose Manual Configuration.
10. Click Apply Settings.
11. Click the tab Consumer security.
12. In the Authentication settings, enter the following:
   - 1. User ID: Enter the user name for the procurement realm. For example, ERP2PTESAP_T.
13. Click the tab Transport Settings and enter the following details:
How to configure client certificates to import transactional data directly

Prerequisites

You must have the client certificate that you have received from an Ariba trusted Certificate Authority.

Context

To ensure that the connection request is originating from a known client, you must configure client certificates when importing transactional data using direct connectivity.

Note

You are not required to perform this procedure if you already use direct connectivity for the master data on the same service.
Procedure

1. Import the private key of the certificate into the SAP Business Site system by using the Trust Manager (transaction code STRUST). You can only import certificates in the Personal Security Environment (PSE) format. Certificates in other formats must first be converted to the PSE format. Use the command line tool SAPGENPSE to do the conversion. Install the SAPGENPSE tool with the SAP Cryptographic Library installation package. For more information, see The SAP Cryptographic Library Installation Package [external document].

   For example, to convert the P12 (Public-Key Cryptography Standards) format to PSE format, enter the following command line:

   sapgenpse import_p12 -v -r <root certificate> -p <Target PSE file><Source File>.

      ○ Enter ARIBA as the identify name and Ariba Network Client as the description.
      ○ Save the entries.
   2. Import the private key of the certificate into the SAP Business Site system by using the Trust Manager (transaction code STRUST).
      ○ Select the newly created ARIBA SSL Client ID and choose PSE Import to import the PSE file.
      ○ Enter the password for the certificate, if required.
      ○ To save, click PSE Save as SSL Client.
      ○ Enter ARIBA as the SSL Client.
      ○ Navigate to the Own Certificate group box on the Trust Manager screen, and double click the certificate to add it to the certificate list. You can now see the newly added certificate in the Certificate List of the Trust Manager.

2. Import the root certificate into the SAP Business Suite system by using Trust Manager as follows:
   1. Double-click the SSL Client Identity ARIBA that you have created.
   2. Navigate to the Certificate group box and choose Import certificate. Click Add to Certificate List to add the imported certificate to the certificate list.
3. Obtain the server certificate from Ariba for the HTTPS SSL encryption:
   2. Download the certificate using your browser:
      For example, if you are using Internet Explorer, go to Internet Explorer, click View Security Report View Certificates. On the tab Details, click Copy to File and export it in the Base-64 encoded X.509 format.
   3. Import the server certificate into the SAP Business Suite system using Trust Manager.
   4. Double click the ARIBA SSL Client ID that you have created.
   5. Navigate to the group box Certificate and choose Import certificate. Add the imported certificate to the certificate list by clicking Add to Certificate List.
4. Run the transaction code SMICM to activate the changes and restart the Internet Communication Manager (ICM). Click Administration ICM Restart Yes. For more information, go to the SAP documentation portal and search for the phrase Using the ICM Monitor.
5. Run the transaction code SOAMANAGER to configure the Web services.
   1. Click Web Service Configuration.
2. On the tab **Design Time Object Search**, search for **MIOut:* consumer proxy**.

3. In the **Search Result** section, click **/ARBA/CO_MIOUT_SYNC_ARIBA_DOWN**.

4. On the tab **Configurations**, click **Create** and then click **Manual Configuration** to create a manual logical port.

5. In the text box **Logical Port Name**, enter a name for the logical port.

6. Enter a description in the text box **Description**.

7. Check **Logical Port is Default**.

8. Click **Next**.

9. On the Consumer Security page, select the **X.509 SSL Client Certificate** and then click **X.509 SSL Client Certificate**.

10. Enter **ARIBA** in the text box **SSL Client PSE of transaction STRUST** and click **Next**.

11. On the HTTP Settings page, enter the **URL Access Path**, **Computer Name of Access URL**, **Port number of Access URL** and **URL Protocol Information**.

   **Note**
   If you choose HTTPS as the **URL Protocol Information**, ensure that you enter **443**. If you choose HTTP, enter **80**.

12. Click **Finish**.

13. Ping the webservice URL. The following error message appears: **Web service ping failed. (RC=403. Service Ping ERROR Forbidden.)**

6. Log in to your Ariba Buyer account and go to **Core Administrator > Integration Tool Kit Security**.

The Integration Toolkit Security page appears.

7. In the pulldown **Select the Authentication Method**, choose **Certificate**. Paste the contents of the public certificate.

## How to run the transactional data integration tasks

### Context

To create transactional data in SAP ERP using either direct connectivity or Ariba Integration Toolkit, you run the following transactional data integration tasks.

### Procedure

1. Run the transaction code **/N/ARBA/OND_PROC** in SAP ERP.

2. Navigate to **SAP menu > Transactional Data > Reports** to expand the tree structure.

3. Double-click the required transactional data report.

4. Do one of the following:
   ○ Click **Ariba Integration Toolkit** to run the transactional data using the Ariba Integration Toolkit or your own tool.
○ Click **Direct Connectivity** to run the transactional data tasks directly.

5. Enter the following configuration for the transactional data:
   ○ **Encoding to be used**: Specify the encoding type or use the default value available. For example, UTF-8.
   ○ **Variant**: Specify the variant for your Ariba Procurement Solution.
   ○ **Partition**: Specify the partition for your Ariba Procurement Solution.

6. Select the transactional data tasks to be exported to the Ariba Procurement Solution.

7. Click **Execute**.

---

### Integrating transactional data using the mediated connectivity integration method

About integrating transactional data using the mediated connectivity integration method [page 250]

Configuring transactional data using the mediated connectivity integration method [page 251]

### About integrating transactional data using the mediated connectivity integration method

Buyers can integrate transactional data directly from the Ariba Procurement Solution to SAP ERP using the SAP Process Integration layer using the Mediated Connectivity integration method. This provides enhanced security while downloading transactional data from Ariba Procurement Solution to SAP ERP using the mediated connectivity.

To integrate transactional data, a buyer can use either client certificates or shared secret authentication to connect to the Ariba Procurement Solution through SAP Process Integration using the **Direct Connectivity** option.

**Note**

You can use the **Direct Connectivity** option for both mediated and direct connectivity.

If you are using the client certificate, the SAP Process Integration keystore stores the certificates and keys that you create in SAP Process Integration. You can create a view to group these certificates and keys based on views.

If you are using the shared secret authentication, you must configure the shared secret in the Communication Channel of your SAP Process Integration.

**Note**

If you are using direct connectivity, then the shared secret is maintained in the logical port.
Prerequisites

You must download and import the transport requests into your SAP ERP system.

Ensure that you do the following:

- Install client certificates if you use the client certificate authentication. You must get a client certificate from Certificate Authorities that are trusted by Ariba. Additionally, you must implement the SAP Note 1669829. This SAP Note is applicable for the following releases:
  - SAP_BASIS release 7.0 lower than SP27
  - SAP_BASIS release 7.01 lower than SP12
  - SAP_BASIS release 7.02 lower than SP11
  - SAP_BASIS release 7.31 lower than SP03
- Do not create the logical port for the consumer proxy in the SOAMANAGER: MIOut_Sync_AribaDownload.
- Implement the following SAP Notes:
  - 1402826 - Supplement to Note 548131
  - 1716777 - Runtime error IMPORT_WRONG_END_POS when displaying class
- Before you export the transactional data, you must create the Authorization Object class.

Limitations

- Administrators do not get email notifications when an error occurs during the transactional data integration.
- Administrators can only view the error messages in the runtime monitor of the SAP ERP and SAP Process Integration.
- The error messages do not store a record of all the error conditions in the transaction SLGI.

Related Information

How to configure client certificates to import transactional data directly [page 247]
How to configure the shared secret authentication [page 254]

Configuring transactional data using the mediated connectivity integration method

Perform the following procedures to configure transactional data using the SAP Process Integration layer:

1. How to install client certificates [page 244]
2. Defining authorizations for running Ariba Procure-to-Pay transactions [page 90]
3. How to define site name in the maintenance table [page 245]
4. How to configure the proxy on SAP ERP systems [page 166]
How to configure the proxy on SAP ERP systems

Procedure

1. Run the transaction code SM59 and check if following configurations exist:
   - SAP_PROXY_ESR under HTTP Connections to External Server
   - XI_INTEGRATIONSERVER_<SID> (SID - system identifier) under HTTP Connections to ABAP System (H Type RFC)
   - LCRSAPRFC under TCP/IP Connections. Specify the program ID similar to the one created under Jco RFC destinations (LCRSAPRFC_<SID>, where SID - system identifier)
   - SAPSLDAPI under TCP/IP Connections. Specify the program ID similar to the one created under Jco RFC destinations (SAPSLDAPI_<SID>, where SID - system identifier)
2. Run the transaction code SLDAPICUST.
3. Enter SAP XI hostname, port, XI user ID, and password.
4. Run the transaction code SXMB_ADM.
5. Navigate to Integration Engine ▶ Configuration.
6. Click Integration Engine Configuration to edit global configuration data.
7. In the Global Configuration Data section, enter the following:
   - Role of Business System: Application System
   - Corresponding Integ. Server: dest://<XI_INTEGRATIONSERVER_PPI>
     This is the same H Type RFC destination mentioned in Step 1.
8. Run the transaction code SLDCHECK.

The SAP Process Integration server appears and the Summary section indicates that the connection to XI is successful.

How to configure the proxy for single stack (Java only) instances for SAP Process Integration 7.3 and higher

Procedure

1. Run the transaction code SM59 and create RFC destination SAP_PROXY_ESR (Type G) under HTTP Connections to External Server.
1. Under the Technical Settings tab, add AEX (PI Java Stack) server host name and port.
2. In the Path Prefix field, enter /rep.
3. Under the Logon & Security tab, you must ensure that the user is assigned to the role SAP_XI_REP_SERV_USER.

2. Create an HTTP destination to AEX of Type G (example: PI_AEX) as follows:
   1. Under the Technical Settings tab, add AEX (PI Java Stack) server host name and port.
   2. In the Path Prefix field, enter /XISOAPAdapter/MessageServlet?ximessage=true.
   3. Under the Logon & Security tab, user must be assigned to role SAP_XI_REP_SERV_USER.

3. Run the transaction code SXMB_ADM.

4. Navigate to Integration Engine Configuration.
5. Click Integration Engine Configuration to edit global configuration data.
6. Under Global Configuration Data, enter:
   - Role of Business System: Application System
   - Corresponding Integ. Server: dest://<PI_AEX>
     This is the same G Type RFC destination mentioned in step 2.
7. Click Specific Configuration and enter the following:
   - Category: Runtime
   - Parameter: IS_URL
   - Current Value: dest://<PI_AEX>
     This is the same G Type RFC destination mentioned in step 2.
8. Run the transaction code SLDAPICUST.
9. Enter SAP XI hostname, port, XI user ID, and password.
10. Run the transaction code SLDCHECK.

---

**How to configure certificates and keys to import transactional data using the SAP Process Integration layer**

**Procedure**

1. Go to the SAP Process Integration main page and click NetWeaver Administration and then click Configuration Management.
2. Click Certificate and Keys.
3. On the Certificate and Keys page, click Create to create a new view. Create the required keys and certificates for the newly created view.

   **Note**
   
   Ensure that your required certificates are signed by an Ariba trusted Certificate Authority to successfully establish the SSL connection.

4. Go the SOAP Receiver Communication Channel page and in the tab General, check Configure Certificate Authentication.
5. In the field **Keystore Entry**, choose the key newly created in the keystore.
6. In the field **Keystore View**, choose the newly created view in the keystore.
7. In the field **Target URL**, ensure that you specify the location of the certificates stored on your SAP Process Integration system.

   *Note*

   Ensure that your URL starts with `https://cert...`

8. Save and activate the communication channel.
9. Log in to your Ariba Buyer account. Go to **Core Administrator** > **Integration Tool Kit Security**. The page **Integration Toolkit Security** opens.
10. In the pulldown **Select the Authentication Method**, choose **Certificate**. Paste the contents of the client certificates.

### How to configure the shared secret authentication

#### Context

You are required to configure the shared secret in the **Communication Channel** of your SAP Process Integration for upload and download communication channels.

You must perform the following procedure separately for download and upload communication channels:

#### Procedure

1. Open the Configuration Integration Builder from the main SAP Process Integration page.
2. In the **Objects** tab of the Integration Builder, click **Communication Channel**.
3. Go to the page **Communication Channel** and click **Configure User Authentication**.
4. In the **Target URL** field, ensure that you specify the location of your procurement system. For example:
   - For upload: `/Buyer/fileupload?realm`
   - For download: `/Buyer/filedownload?realm`
5. In the text box **User**, enter the realm name for your procurement system.
6. In the text box **Password**, enter the same shared secret that you specified on the **Integration Toolkit Security** page of your Ariba Procurement Solution.
7. If you are using the HTTPS SSL encryption, you must obtain the server certificate from Ariba. To obtain the server certificate, do the following:
   2. Download the certificate using your browser. For example, if you are using Internet Explorer, go to Internet Explorer, click **View** > **Security Report** > **View Certificates**. On the tab **Details**, click **Copy to File** and export it in the Base-64 encoded X.509 format.
3. Import the server certificate into the SAP Process Integration keystore view.

How to configure the integration scenario

Context

To establish a connection between SAP ERP and Ariba Procurement Solution when using Mediated Connectivity (SAP Process Integration), you need two communication channels. One communication channel for download and another for upload. This is a one-time configuration that is required to send and receive all document transactions using the Mediated Connectivity integration method.

To create the communication channels, you must perform the following procedure separately, for download and upload:

Procedure

1. From the SAP Process Integration main page, choose Integration Directory ➤ Integration Builder ➤ Login to the Configuration: Integration Builder application.

2. In the Integration Builder page, choose Tools ➤ Apply Model from ES Repository ➤.

3. In the first screen of the wizard Transfer Model from ES Repository, choose ES Repository Model and then click Process Integration Scenario.

4. Click the field Name and then click the pulldown Display Input Help and then click Input Help.

5. From the screen Select Process Integration Scenario from Enterprise Services Repository, choose the required option:
   - For download: AR_Ariba_Download
   - For upload: AR_Ariba_Upload

6. Click Apply. The wizard Transfer Model from ES Repository appears. The Namespace and the Software Component Version is available in the wizard.

7. Click Continue and then click Close. This completes the configuration scenario for the selected option:
   - AR_Ariba_Download for download
   - AR_Ariba_Upload for upload

8. In the Model Configurator, click Select Component View.

9. In the Select Component View for Configuration page, choose the tab Business System Components for A2A to assign components for your Ariba procurement solution.

10. Click the Insert Line button (+) to add a component entry in the table if no component entry line is present.

11. Click the row under the Communication Component column, then click the Value List drop down and choose Value List ..F4. From the selection list, choose the required communication component name and click Apply:
   - AR_Ariba_Download for download
   - AR_Ariba_Upload for upload

12. Click Assign Component.
13. In the Model Configurator page, choose the Business System Components for A2A tab to assign the communication component for your Ariba Procurement Solution.

14. Click the row under the column Communication Component, then click the pulldown Value List and choose Value List <F4>. Choose the communication component name from the selection list and click Apply.

15. Click the button Next Application Component (the right arrow) to go to the next screen in the window.

16. Click the button Insert Line (+) to add a component entry in the table if no component entry line is present.

17. Click the row under the column Communication Component, then click the pulldown Value List and choose Value List <F4>. Choose the communication component name for the business system from the selection list and click Apply.

18. Click the button Save Settings to save the changes.

19. In the page Model Configurator, click Configure Connections. You must assign a Communication Channel to set up the connections.

20. In the tab Connections from Component Assignment tab, click the empty cell Communication Channel corresponding to the to the business system component value.

21. Click the pulldown Value List and choose Value List <F4>. Choose the communication component and click Apply.

22. Choose the communication channel CC_Proxy_Sender from the selection list and click Apply.

**Note**

You require the Proxy Sender communication channel only for a single stack Java machine. If you are using a dual stack Java machine, the cell Communication Channel must be left blank.

23. Click the button New Communication Channel (the icon at the top left) on the Communication Channels page to start the Create Communication Channel wizard.

24. Click Continue and in the next page click the field Name, then click the dropdown Display Input Help and choose Input Help. Choose the following Communication Channel Template:

- Communication Component: Specify the communication component.
- Communication Channel: Specify the communication channel.

25. Click Finish.

The Model Configurator window appears.

26. In the Model Configurator, click Create Configuration Object.

27. In the window Create Configuration Object, do the following:

- Click Generation in the General section.
- Check Receiver Determination and Interface Determination in the Scope of Generation section.

28. Click Create New in the Change List for Generated Objects section. The text box already contains an object name that you can use or modify.

29. Click Start to begin generating objects.

30. After the object generation is complete, click Apply to save the new configuration settings. Close the log that is created without saving. You can save it, if required.

31. On the page Communication Channel, you can specify the shared secret or client certificate authentication in the tab Parameters. Ensure that you set the parameters in the section General > Connection Parameters that matches your Ariba Buyer account configuration.

32. Save the changes.
**Note**

The proxy settings and the subsequent reflection of object `Ariba_Download ESR` or `Ariba_Upload ESR` in SAP ERP transaction `SPROXY` is done for integrating transactional data through the SAP Process Integration layer. By default, the connectivity is always direct through `SOAMANAGEr` or `LPCONFIG`. 
Error Handling and Troubleshooting

About Error Handling and Troubleshooting [page 258]
Application Errors [page 258]
System Errors [page 261]
Viewing the Application Log [page 264]
Troubleshooting [page 265]
Enabling Debugging [page 266]

About Error Handling and Troubleshooting

The SAP integration provides reliable data exchange between Ariba Procurement Solution integrated with SAP. The integration is designed to recover automatically whenever possible with minimal intervention. The integration event can recover automatically from any error that occurs before the data has been committed to the SAP system. After the data has been committed on the SAP system, the integration event does not try to recover from errors. The integration event reports the error and waits for an administrator to intervene. However, in some cases, the integration event on the Ariba Procurement Solution cannot recover automatically and must report the error to the appropriate administrator. When an error occurs, the integration event on the Ariba Procurement Solution does the following:

- Recover automatically
- Report the error and wait for an administrator to take action

Errors can occur when you integrate master data and transactional data, directly using the Direct Connectivity integration method, the File Channel, or Web Service Channel. The following are the two types of errors that can occur while integrating master data and transactional data:

- Application Error
- System Error

Application Errors

An application error occurs when there are inconsistencies in the data that is being exported to the Ariba Procurement Solution. Ariba recommends that you ensure that the data already exists before you export either from your ERP system or the Ariba Procurement Solution.

Application Errors When Importing Master Data Using Direct Connectivity [page 259]
Application Errors When Exporting Transactional data Using the File Channel [page 259]
Application Errors When Exporting Transactional Data Using the Web Service Channel [page 260]
Application Errors When Importing Master Data Using Direct Connectivity

When exporting the master data, the data is posted on the Ariba Procurement Solution system. At that point, data validation does not occur. Ariba Procurement Solution system validates the data later. When there are data dependencies or inconsistencies, an error is recorded on the Ariba Procurement Solution. Administrators can check email errors through the following methods:

- Configure email notification alerts on their Ariba Procurement Solution system to receive notification when an error occurs. This enables the administrator to log in to the Ariba Procurement Solution to check the details of the errors.
- Log in to the Ariba Procurement Solution and check the details of the errors.

To correct the error, the administrator must ensure that the data inconsistency is resolved in ERP ERP system and then must run the tasks for the master data again from your ERP ERP system.

Application Errors When Exporting Transactional data Using the File Channel

Application Errors During Purchase Order Exports [page 259]
Application Errors During Receipt Exports [page 260]
Application Errors During Payment Exports [page 260]

Application Errors During Purchase Order Exports

The SAP program that reads and populates the exported data in the SAP system can fail due to a data error. In such events, the program generates an error file on the ERP ERP system. This error file creates an error object (PurchaseOrderError) on uploading to the Ariba Procurement Solution system and the requisition status changes to “Composing.”

When a requisition is stuck in the “Composing” state, the administrator for Ariba Procurement Solution must perform the following tasks:

- Check the error in the requisition is in the “Composing” state
- Diagnose the problem
- Correct the error
- Submit or approve the requisition

Errors in SAP are logged as .csv files (for example, PurchaseOrderErrorPullXXXX.csv). These files are transferred to the Ariba Procurement Solution through an import integration event. The Ariba Procurement Solution creates the error objects and the order status changes accordingly.
Application Errors During Receipt Exports

The Receipts File export event can be run manually or scheduled through the Data Transfer Tool. This event exports the data into CSV files.

After the items are received in the Ariba Procurement Solution, the ProcessReceipts schedule task must be run. This scheduled task picks up all receipts in the “Awaiting processing” state and adds the entries in PushDataEntry.

If an application-level error occurs during the receipt export (that is, the posting period is not open in SAP), the SAP ERP system rejects the receipt. The SAP ERP system also generates an error file. After this error file is uploaded to the Ariba Procurement Solution, the receipt status changes from "Processing" to "Awaiting processing", and the subsequent ProcessReceipts task processes the receipts.

Application Errors During Payment Exports

When invoices are reconciled, payment objects are generated and entries are added to the PushDataEntry. Administrators can run the PaymentExport task and download the data into CSV files on the ERP ERP system

While processing the payment files and an application error occurs. This usually occurs when there is an invalid supplier or a purchase order is already under process or is being processed by another user. In such a case, the ERP ERP system creates an error file.

When the administrator uploads this error file to the Ariba Procurement Solution, it creates the PayablePushError objects and the status changes to "Paying Failed". Administrators can correct the data and submit again. This data is processed again when the next export event is run.

Application Errors When Exporting Transactional Data Using the Web Service Channel

Application Errors Using Web Service Channel for Purchase Order Exports [page 260]
Application Errors Using Web Service Channel for Receipt Exports [page 261]

Application Errors Using Web Service Channel for Purchase Order Exports

An Ariba Procurement Solution always initiates a purchase order export with an integration event that exports the order out to the Web Services integration channel.

The most common error is when the SAP ERP system rejects the incoming order because it contains data that is not valid for the SAP ERP system. For example, if a supplier is invalid in the SAP ERP system after the order has been created in the Ariba Procurement Solution, the incoming order is rejected because it has an invalid supplier. For purchase orders, the export event returns error messages and details about exports immediately. When an SAP ERP system rejects an order, it sends that order back with an error message indicating why the order has been rejected.
If an error occurs during the integration event of the real-time purchase order export through Web Services and NetWeaver PI is initiated, Ariba Procurement Solution logs an error, and the purchase requisition associated with that order returns to the “Composing” state.

When a requisition is stuck in the “Composing” state, the Ariba procurement solutions administrator should perform the following tasks:

- Check the error in the requisition that is in the “Composing” state
- Diagnose the problem
- Correct the error
- Submit and approve the requisition

**Application Errors Using Web Service Channel for Receipt Exports**

If an application-level error occurs during the receipt export (that is, the posting period is not open in SAP), the SAP ERP system rejects the receipt. The processing state of the receipt object in Ariba procurement solutions changes from “Processing” to “Awaiting Processing”. Ariba Procurement Solution attempt to export the receipts again when the ProcessReceipts scheduled task is run.

**System Errors**

A system error occurs when there are network errors due to an incorrect URL address or when the connection to the Ariba Procurement Solution is lost.

- System Errors When Importing Master Data Using Direct Connectivity [page 261]
- System Errors When Exporting Transactional Data Using Direct Connectivity [page 262]
- System Errors When Exporting Transactional Data Using the File Channel [page 263]
- System Errors When Exporting Transactional Data Using the Web Service Channel [page 263]

**System Errors When Importing Master Data Using Direct Connectivity**

System error can occur either when data is sent from the ERP system to the PI or from the PI to the NetWeaver system. An administrator can monitor the errors from the SXMB_MONI if a system error occurs while sending the data from the ERP system to the PI. The monitor of the NetWeaver system displays the error when the data is sent from the PI to the NetWeaver system.

When a system error occurs, the ABAP program tries to upload the master data up to three times based on the time delay specified in the /ARBA/AUTH_PARAM table. If it still fails, the master data is not lost and is available on the SAP ERP server. To retrieve the master data, the administrator must make the necessary changes to the configuration (RealmID, shared secret, or URL) based on the error and run the Full or Incremental Load exports for the master data tasks again from your ERP system.
When exporting the master data, the data is posted on the Ariba Procurement Solution system. The data validation does not occur immediately on posting the data but later. When there are data dependencies or inconsistencies, an error is recorded on the Ariba Procurement Solution.

Administrators can check errors using the following methods:

1. Configure email notification alerts on their Ariba Procurement Solution system to receive notification when an error occurs. This enables the administrator to log in to the Ariba Procurement Solution to check the details of the errors.
2. Directly log in to the Ariba Procurement Solution and check the details of the errors.

To correct the error, the administrator must ensure that the data inconsistency is resolved in ERP system and then must rerun the tasks for the master data again from the ERP system. The administrator must rerun the Full or Incremental Load for the master data tasks again from your ERP system.

**Note**

Buyers can also manually run the master data import tasks.

### Single Stack

To view errors in a single stack instance, go to the Configuration and Monitoring Home>Adapter Engine>Message Monitor page. You can also view errors faced when integrating master data using the Mediated Connectivity integration method.

### Dual Stack

To view errors in a dual stack instance, go to the SXMB_MONI page. You can also view errors faced when integrating master data using the Mediated Connectivity integration method.

### System Errors When Exporting Transactional Data Using Direct Connectivity

An administrator can monitor the errors from the SXMB_MONI, of the SAP ECC system or SAP Netweaver PI, if a system error occurs while sending the data from the SAP ERP system to the SAP Netweaver PI. The monitor of the SAP Netweaver PI system displays the error when the data is sent from the SAP Netweaver PI to the Ariba Procurement Solution.

When a system error or application error occurs, timestamp is not updated in the /ARBA/DATE_TIME table. As a result, the transactional data created in the Ariba Procurement Solution does not change its status. The administrator must correct the errors in the SAP ERP and then rerun the relevant report.

When any data error occurs, timestamp is updated in the /ARBA/DATE_TIME table. However, the administrator must correct the errors in the Ariba Procurement Solution and then rerun the relevant report. If there are any data
errors when creating the transaction data, the corresponding status is uploaded to the Ariba Procurement Solution.

System Errors When Exporting Transactional Data Using the File Channel

When exporting transactional data using the File channel through the Ariba Integration Toolkit and an error occurs, the error is recorded in the log files.

In case of a configuration error, or errors in the Ariba Integration Toolkit, SAP never receives any transactional data from Ariba Procurement Solution. In such situations, all transactional objects remain in the same state and will not change.

When there is a system level error the administrator must try to export the data again from Ariba Procurement Solution using the time stamp from that you need the data. If the CSV file that is exported to the SAP ERP system is not picked up by the SAP program, these files remain on the SAP server, and are picked up the next time the SAP program is run. For additional information, see the Ariba Procurement Data Import and Administration Guide.

System Errors When Exporting Transactional Data Using the Web Service Channel

System Error During the Purchase Order Export [page 263]
System Error During the Receipt Export [page 264]

System Error During the Purchase Order Export

If there is a network issue or the NetWeaver PI is down, the SAP ERP system cannot receive a purchase order from Ariba Procurement Solution. In this case, the purchase orders or requisitions are stuck in the “Ordering” state.

Once a system-level error has been found, nothing is required to be done to restart the export task. The scheduled task FailedOrders checks for the unfinished orders and executes the order transmission again. As an administrator, you can either wait until the next scheduled run of this task, or execute it manually from Ariba Administrator.

The ProcessStuckERPOrders scheduled task, is not run on a regular schedule but is run manually by an administrator to correct the error. This task changes the state of purchase orders, change orders, and cancel orders in the SAP ERP system to “Ordering/Failed” so that the FailedOrders scheduled task can attempt to process them again. For example, an order might get stuck in “Ordering” if a network problem or web service channel error occurs and the SAP ERP system does not receive the order or Ariba Procurement Solution does not receive the status.
System Error During the Receipt Export

If a system level error occurs during the receipts export, the receipts remain in the "Processing" state. The administrator must run the ProcessPendingReceipts schedule task. This changes the status from "Processing" to "Awaiting Processing", and the receipts are picked up by the ProcessReceipts task.

For more information on scheduled tasks, see the Ariba Procurement Data Import and Administration Guide.

Viewing the Application Log

Context

To determine if the connection between the SAP ERP system and Ariba Procurement Solution succeeded or failed, and the reason for failure, you can view the application log using the following procedure.

Procedure

1. Go to transaction code SLG1.
   The Analyze Application Log appears.
2. Enter /ARBA/TR_OBJECT in the Object field.
3. Enter any of the following required sub-object in the SubObject field:

<table>
<thead>
<tr>
<th>Sub-Object Name</th>
<th>Sub-Object For Ariba Procurement Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/PURCHASEORDER</td>
<td>Purchase Orders</td>
</tr>
<tr>
<td>/ARBA/CHANGEORDER</td>
<td>Change Purchase Orders</td>
</tr>
<tr>
<td>/ARBA/CANCELORDER</td>
<td>Cancel Purchase Orders</td>
</tr>
<tr>
<td>/ARBA/POHEADER</td>
<td>Purchase Order Header Status</td>
</tr>
<tr>
<td>/ARBA/GOODS_RECEIPTS</td>
<td>Goods Receipts</td>
</tr>
<tr>
<td>/ARBA/REMITTANCE</td>
<td>Remittances</td>
</tr>
<tr>
<td>/ARBA/INVOICE</td>
<td>Invoices</td>
</tr>
<tr>
<td>/ARBA/EXPENSE</td>
<td>Expenses</td>
</tr>
</tbody>
</table>
Troubleshooting

This section provides tips for solving problems associated with integrating Ariba Procurement Solution integrated with SAP.

The “Internal Error” stack traces that you may encounter do not cause error conditions and can be safely ignored. Furthermore, your SAP events run successfully while or after these traces appear.

Failures When Importing Remittances from SAP ERP System

When importing remittances, an error can occur due to an application or system error.

Errors while importing remittances using the Web Service channel

When remittances fails during the import, the date and time stamp must be changed in the OverrideStartDate parameter on the Ariba Procurement Solution and the Incremental Remittance Import task must be run again. Administrators can run the task manually or schedule it to run.

Errors while importing remittances using the File channel

When remittances fail during the import, the administrator must change the date and time stamp in the /ARBA/REM_DATETM table and run the RemittanceImport program again. This task can be scheduled or run manually.

Error condition when you re-import the RFC schema for requisition, purchase order, invoice, or derivation into the SAP NetWeaver PI and you use the RFC schema for mapping:

Error Message:

The source structure, target structure, or a function library has been changed or could not be found in the Enterprise Services Repository. The mapping definition contains elements or attributes that do not exist in the changed structure, or functions that were changed in a function library. The relevant entries will be deleted.

Target structure has no such path: /ns1:_-ARBA_-DERIVATION/CHECK_CODINGBLOCK/item/BUDGET_PERIOD.

Skipping mapping

Cause: This error occurs when you re-import the RFC schema for requisition, purchase order, invoice, or derivation into the SAP NetWeaver PI and you are using:
- SAP ERP EHP version 4.0 and above and you have not activated the budget period business function
- SAP ERP EHP version lower than 4.0

**Solution:** Perform the following steps when the error message appears:

1. Click **Close**.
2. On the **Definition** tab of the **Display Message Mapping** screen that appears, click the **Correct Structural Inconsistencies** button (extreme right).
3. In the **Move Field** screen that appears, click **Skip**.
4. Click **Save** and activate.

**Troubleshooting integration of service procurement in the Ariba Procurement Solution integrated with SAP ERP**

**Error Condition:** You cannot edit service orders in the Ariba Procurement Solution and you receive an error message stating that the service order is either being edited or could not be sent to the ERP system.

**Cause:** This occurs when the service order is open in SAP ERP.

**Solution:** Close the service order in SAP ERP.

**Related Information**

[About integration of service procurement in the Ariba Procurement Solution integrated with SAP ERP][234]

---

**Enabling Debugging**

When you call an RFC, you can turn on a debugging switch. This action enables you to access the SAP user interface, where you can set breakpoints.

**Note**

To debug File channel integration programs, you can use the standard SAP ABAP debugging functionality.
Debugging Using the SAP User Interface

Procedure

1. Go the transaction SE38 and open the program that you want to debug, in the display mode.
2. Put the break point at the required code statement.
3. Execute the program. ABAP debugger stops at the break point that you have set.

Debugging Parameters in Receipts

Ariba Procurement Solution RFCs offer the /ARBA/GR_PUSH_DISPLAY_MODE parameter. To view export tasks, the administrator must change the parameters from 'N' to 'E' or 'A'. Changing these parameters also enables you to view goods receipt screens in debug mode.
Account Assignment Categories

The material in this chapter is appropriate for SAP administrators and developers and Ariba procurement solutions implementation personnel.

About Account Assignment Categories [page 268]
Configuring Account Assignment Categories [page 269]

About Account Assignment Categories

SAP uses account assignment categories to specify how to assign a line-item expenditure to an account. Examples of account assignment categories are assets, WBS elements, or cost centers. Account assignment categories determine what information a user must provide for a line item, such as the account number of an asset.

For example, if you were to purchase an elevator for your company’s building through Ariba procurement solutions, you could apply the cost of the elevator through an asset account category. You might do so because buildings are usually considered assets. Quite often, special tax and accounting considerations exist for expenses related to the improvement or maintenance of assets.

Such considerations might make it necessary for your SAP system to handle expenses for assets in a different manner than expenses for other items such as manufacturing raw materials or office supplies. These differences affect the type of information SAP displays to users who enter asset line items, and what they must do with this information. Account assignment categories let SAP know how to handle these differences.

For example, if a user chooses an asset account assignment category, SAP does not require the entry of any information related to cost centers. When possible, SAP includes default values for required information based upon current data.

It is possible to configure all existing SAP account assignment categories to display and require the accounting fields you want. If the default account assignment categories do not suit your needs, you create your own categories.

Account Assignment Category Imports

Ariba Procurement Solution imports account assignment categories from SAP. The import from SAP includes information about how the Ariba Procurement Solution user interface must treat account assignment categories to know:

- What types of accounts to display in the user interface.
- Whether to require the user to choose or provide certain types of account information in the user interface.

The default Ariba Procurement Solution SAP configuration imports the categories listed in the following table. The table identifies which entries are required for each assignment category. The requirement settings for these entries are fixed; the entries are required regardless of the values in the FAUS1 field in AccountCategory.csv.
### Configuring Account Assignment Categories

Ariba procurement solutions imports account category information and uses it in similar ways to those used by SAP. This section explains how SAP allows you to configure account assignment categories, and then explains how Ariba procurement solutions imports, displays, and handles account assignment categories in the user interface.

### Understanding Ariba Procurement Solutions Accounting Types

Ariba procurement solutions defines a generic accounting entity, which is a piece of accounting information. Examples of accounting entities include Project, Cost Center, Account, and SubAccount.

In a typical configuration, each site defines its own set of accounting entities. The accounting fields are not extrinsic—they are just specific instances of the generic accounting entity.

The default Ariba procurement solutions configuration for SAP defines the following accounting entities:

- asset—Asset
- Sub Asset—Sub Asset
- cost center—Cost Center
- general ledger—G/L Account
- internal order—RK/PPC order
- WBS element—Project

Ariba procurement solutions imports appropriate data from your SAP system and uses that data to populate the choosers and accounting information in Ariba procurement solutions. Ariba procurement solutions imports and uses SAP account assignment category information to determine what users need to enter for a line item.

---

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Assignment Category</th>
<th>Required Entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Asset</td>
<td>Asset Number</td>
</tr>
<tr>
<td>F</td>
<td>Order</td>
<td>Internal Order, GL Account, Asset, and Sub Asset</td>
</tr>
<tr>
<td>K</td>
<td>Cost Center</td>
<td>Cost Center, GL Account</td>
</tr>
<tr>
<td>P</td>
<td>Project</td>
<td>WBS Element, GL Account</td>
</tr>
</tbody>
</table>

**Note**

In the default configuration, the **CostCenter** field is not displayed on the line-item details screen when the user chooses account assignment category “F” (Order).

**Note**

The `/ARBA/MASTER_DATA_EXPORT` program does not extract statistical WBS elements.
When setting up your Ariba procurement solutions configuration, you define the accounting entities you plan to use. You also set up your configuration to default accounting information for catalog items from commodity codes. For information on how to configure accounting information in Ariba procurement solutions, see the Ariba Procurement Data Import and Administration Guide.

Configuring SAP Accounting Fields

Context

SAP has a set of information fields it can display or require for any account assignment category. You control the way in which the SAP user interface displays a field and how a user can modify it.

Procedure

1. To control how SAP displays fields and how users modify them, choose one of the following four options:
   ○ Required Entry—SAP displays the accounting field and requires user input. (This setting is ignored for some account category entries. See Account Assignment Category Imports [page 268] for a list.)
   ○ Optional Entry—SAP displays the accounting field, and a user can change it, but SAP does not require the user to make any changes.
   ○ Display—SAP displays the accounting field, but the user cannot change it.
   ○ Hidden—SAP does not display the accounting field.

2. To maintain these properties for each account assignment category, choose Go to Details in transaction OME9 in the SAP user interface.

Example Account Assignment Category Settings

The default asset account assignment category settings within an International Demonstration and Education System (IDES) are as follows:

- Display and require the user to enter an asset name or ID.
- Display but do not require the user to enter the following:
  ○ Asset sub-number
  ○ Business area
  ○ Deletion indicator
  ○ Details account assignment block
  ○ Quantity/Percent for material account assignment
  ○ Goods recipient/Ship-to party
  ○ Unloading point
  ○ Special region
- Display the following for informational purposes, but do not allow the user to change:
○ General ledger account
○ Project
○ CO/PP order
• Do not display the remaining fields.

Understanding Field Status Group Strings

SAP stores information about when to display or require user input for account assignment categories in field status group strings. SAP uses these strings to store the settings previously discussed in the Change View “Account Assignment Categories”: Details screen, in transaction OME9.

One field status group string exists for each account assignment category in your SAP instance. The strings reside in table T162K, within field faus1. Ariba procurement solutions imports the strings and interprets them to handle accounting fields within the user interface.

Each string consists of a series of four characters or spaces. Each of the four characters corresponds to the four settings SAP allows for accounting fields for a given account assignment category. Ariba procurement solutions treats some settings differently from SAP within the user interface.

<table>
<thead>
<tr>
<th>Character</th>
<th>SAP User Interface</th>
<th>Ariba procurement solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘+’ (plus sign)</td>
<td>Required entry</td>
<td>Displays the field and requires user input.</td>
</tr>
<tr>
<td>‘.’ (period)</td>
<td>Optional entry</td>
<td>Displays the field, and the user can decide to whether to change it.</td>
</tr>
<tr>
<td>‘*’ (asterisk)</td>
<td>Display</td>
<td>Does not display this field because it does not require user input.</td>
</tr>
<tr>
<td>‘-’ (minus sign)</td>
<td>Suppressed</td>
<td>Does not display this field because it is not relevant.</td>
</tr>
</tbody>
</table>

The position of each character within the string determines the accounting field to which it applies. For example, in an IDES system, the field status group strings for the default Ariba procurement solutions account assignment categories are as follows:

<table>
<thead>
<tr>
<th>Account Type</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost Center</td>
<td>10</td>
</tr>
<tr>
<td>Internal Order</td>
<td>11</td>
</tr>
<tr>
<td>WBS Element</td>
<td>12</td>
</tr>
<tr>
<td>Asset</td>
<td>24</td>
</tr>
<tr>
<td>General Ledger</td>
<td>51</td>
</tr>
</tbody>
</table>

For example, the positions for an asset category from an IDES system resemble those in the following diagram:
Ariba procurement solutions interprets this field status group string in the following way:

The Cost Center character is a minus sign, which means it is not relevant to asset line items and does not appear in the Ariba procurement solutions user interface.

The Internal Order, WBS Element (project), and General Ledger characters are asterisks. Ariba procurement solutions does not display them because it does not require the user to enter a value.

The Asset character is a plus sign. Ariba procurement solutions displays the asset field in the user interface and requires the user to enter a value.

Ariba procurement solutions imports the entire field status group, even though the default configuration examines only five of the character positions. Ariba procurement solutions does so in case you want to customize Ariba procurement solutions to use and display other accounting type objects.

Your SAP sites create objects with unique names. The file ariba/variants/variant/partitions/partition/data/FieldStatusToAccountingFieldNameMap.csv contains mappings of field positions to the account type. Make sure to check this file for the new account type you added.

If you implement your own SAP accounting objects to import other accounting categories, the status strings are available for you to use. You can modify the RFCs that pull account categories with selection filters in /ARBA/TVARV to include categories for custom accounting objects.

The implementation of a new accounting object in Ariba procurement solutions is beyond the scope of this document, but generally includes creating a new ClusterRoot object that resembles the existing accounting types.

### Viewing Field Status Group Strings

#### Context

It is possible to view the field status group strings that the Ariba procurement solutions RFC imports for account assignment categories on your SAP instance.
Procedure

1. Run transaction SE37 to access the Function Builder Library: Initial Screen.
2. Type */ARBA/ACCOUNT_CATEGORY_EXPORT* in the Function module field, and click Sngl.test or select
   Function mode ➔ Test ➔ Single test ➔
   The Test Function Module Initial Screen appears.
3. In the Value column, enter your Ariba procurement solutions variant name and partition, and then click the
green check mark button to execute.
   For example, enter vsap and sappart1.

Importing New Accounting Objects From SAP

Context

You might want to configure account assignment imports to provide display information for a new accounting
object you have created in Ariba procurement solutions.

Procedure

1. Within SAP, set up a parameter to an entry in */ARBA/TVARV* for the account assignment category RFC, */ARBA/ACCOUNT_CATEGORY_EXPORT*.
   In the default configuration, use the showField method to validate custom account categories with their field
   status strings.
2. The default behavior of displaying these fields is based on the visibility conditions in the code that follows this
task. If you want other behavior, you could add a cluster root object in the metadata XML visibility condition
calls.
3. For ideas about how to validate custom account assignment categories with their field status strings, examine
the code that follows this task. For example, you could use the showField, within a metadata XML user
interface description.

The comments provide a good summary of the showField method:

```/**
* Given a field on Accounting for the SAP variant, and a given
* account assignment category on that Accounting object,
* determine if the accounting field should show up. This is
* based on SAP's account category property called FieldStatus
* Group. We pull this field status group back from SAP when we
* pull AACs (Account Assignment Categories. The field status
* group is a string of dots, slashes, plusses, and stars.
* Each of these means a different thing.
* 'Optional' ('.') means the accounting field should be shown,
* but does not require any kind of input.
* 'Suppressed' ('-') means the accounting field should not be
* shown at all.
* 'Required' ('+') means the accounting field requires input,
```
and therefore it should be shown. 'Display' ('*') in SAP means visible, but no input possible.
In Ariba that means that the field is simply not visible. SAP has this option 'display' for when it defaults values to an
accounting field and wants the field to be visible for informational purposes only. Since we don't do any defaulting
like that, it's not necessary to show such a field.

```java
public static boolean showField (String fieldName, ValueSource aac)
{
    if (null == aac) {
        return false;
    }
    String Unique_Name = (String)aac.getFieldValue("UniqueName");
    String fstag = (String)aac.getFieldValue("FieldSelectionString1");
    char code = fstag.charAt(positionForField(fieldName));
    /**For an optional field, if the following combinations of Account
     Category (InternalOrder) and Accounting Field (CostCenter) are
     encountered, then we will suppress the display of the CostCenter field.*/
    if ('.' == code){
        if (Unique_Name.equals("F") ) {
            if ( fieldName.equals("CostCenter") ) {
                return false;
            }
        }
        //Optional
        return true;
    }
    else if ('-' == code) {
        // Suppressed
        return false;
    }
    else if ('+' == code) {
        // Required
        return true;
    }
    else if ('*' == code) {
        // Display-only
        return false;
    }
    else {
        // error handling
        return false;
    }
}
```
Integration Events Reference

This appendix lists the integration events provided with Ariba procurement solutions that work with SAP configurations on the Web Services and File channels. For complete information on these integration events, see the Ariba Procurement Data Import and Administration Guide.

Common Integration Events

The following table provides a comprehensive list of all integration event names. This list includes integration events for both File as well as Web Services channel.

**Note**

All master data integration events are performed using the File Channel, while transactional data integration can be performed using either File or Web Services, with Web Services being the default channel.

In the “SAP Integration Events” table, all transactional data integration events specific to File Channel are indicated explicitly. For example, PaymentFileExport (File Channel).

### Integration Events

#### Table 10: SAP Integration Events

<table>
<thead>
<tr>
<th>Event Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AccCategoryFieldStatusComboPull</td>
<td>Imports combinations of Account Category and Field Status codes, used in Ariba Procurement Solution to simulate accounting combinations used in SAP.</td>
</tr>
<tr>
<td>AccountCategoryLanguagePull</td>
<td>Defines translations for the names of Account Categories.</td>
</tr>
<tr>
<td>AccountCategoryPull</td>
<td>Defines the accounting entity AccCategory, used in the default configurations.</td>
</tr>
<tr>
<td>AccountTypeLanguagePull</td>
<td>Defines translations for the names of Account Types from manually maintained CSV files.</td>
</tr>
<tr>
<td>AssetPull</td>
<td>Defines the accounting entity Asset, used in the default configurations.</td>
</tr>
<tr>
<td>BuyerPaymentBankLocationPull</td>
<td>Reads payment bank information for the buying company from a CSV file. For more information, see the Ariba Invoicing and Payment Data Import and Administration Guide.</td>
</tr>
<tr>
<td>Event Name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CombinedDataPull</td>
<td>Performs a batch data load for all master data events. Language pulls can also be run in batch and also part of the CombinedDataPull.</td>
</tr>
<tr>
<td>CommodityExportMapPull</td>
<td>Defines a mapping from commodity codes to SAP-specific ones. Data is loaded from manually maintained CSV files.</td>
</tr>
<tr>
<td>CompanyCodeIOComboPull</td>
<td>Defines combinations of company codes and internal orders, used in default configurations.</td>
</tr>
<tr>
<td>CompanyCodePull</td>
<td>Defines the SAP accounting entity CompanyCode.</td>
</tr>
<tr>
<td>CompanyCodeWBSComboPull</td>
<td>Defines combinations of company codes and WBS elements, used in default configurations.</td>
</tr>
<tr>
<td>CostCenterLanguagePull</td>
<td>Defines translations for the names of Cost Centers.</td>
</tr>
<tr>
<td>CostCenterPull</td>
<td>Defines the accounting entity CostCenter.</td>
</tr>
<tr>
<td>CurrencyConversionRatePull</td>
<td>Loads the currency conversion rates.</td>
</tr>
<tr>
<td>CostCenterIdocImport</td>
<td>Performs real-time cost-center integrations.</td>
</tr>
<tr>
<td>FieldStatusCodeAccountingFieldMapPull</td>
<td>Maps SAP field status strings to accounting field names.</td>
</tr>
<tr>
<td>GeneralLedgerLanguagePull</td>
<td>Defines translations for General Ledger Pulls. Data is loaded from manually maintained CSV file.</td>
</tr>
<tr>
<td>GeneralLedgerPull</td>
<td>Defines the SAP accounting entity GL (General Ledger).</td>
</tr>
<tr>
<td>GlobalCompanyCodeMapPull</td>
<td>Uses a mapped type to map between local and global company code data within the partition.</td>
</tr>
<tr>
<td>InternalOrderPull</td>
<td>Defines the SAP accounting entity IO (InternalOrder).</td>
</tr>
<tr>
<td>ItemCategoryPull</td>
<td>Defines the kinds of orders that Ariba Procurement Solution can generate and push into your SAP ERP system. Data is loaded from manually maintained CSV file.</td>
</tr>
<tr>
<td>PaymentRealTimeExport</td>
<td>Exports payment information using the Web Services channel.</td>
</tr>
<tr>
<td>PaymentExport (File Channel)</td>
<td>Exports payment information using the File channel.</td>
</tr>
<tr>
<td>PartitionedCommodityCodePull</td>
<td>Loads ERP commodity codes from SAP ERP.</td>
</tr>
<tr>
<td>PartitionedCommodityCodeLanguagePull</td>
<td>Defines translations for ERP commodity code names.</td>
</tr>
<tr>
<td>PlantPull</td>
<td>Defines a set of “plants,” which is a concept similar to addresses. Data is loaded from manually maintained CSV file.</td>
</tr>
<tr>
<td>PlantPurchaseOrgComboPull</td>
<td>Imports combinations of plants and purchasing organizations, used in Ariba Procurement Solution to simulate SAP ERP accounting combinations.</td>
</tr>
<tr>
<td>PurchaseGroupPull</td>
<td>Defines a set of PurchaseGroups, by reading from SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrderCancelErrorImport (File Channel)</td>
<td>Imports canceled purchase order error data from SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrderCancelExport (File Channel)</td>
<td>Exports the canceled purchase order data from to SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrderCancelNumberImport (File Channel)</td>
<td>Imports the successfully canceled purchase order data from SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrderErrorImport (File Channel)</td>
<td>Imports the purchase order error information from SAP ERP.</td>
</tr>
<tr>
<td>Event Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PurchaseOrderExport (File Channel)</td>
<td>Exports the purchase order data from SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrderNumberImport (File Channel)</td>
<td>Imports the purchase order status from SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrderChangeErrorImport (File Channel)</td>
<td>Imports changed purchase order error information from SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrderChangeExport (File Channel)</td>
<td>Exports the changed purchase order data to SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrderChangeNumberImport (File Channel)</td>
<td>Imports the successfully changed purchase order data from SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrderCancelRealTimeExport</td>
<td>Exports canceled purchase orders to SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrderChangeRealTimeExport</td>
<td>Exports changed purchase orders to SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrderERPHeaderStatusImport</td>
<td>Imports the SAP ERP header status for a set of Ariba Procurement Solution purchase orders that are in an Ordered state and that are created during a specified time period.</td>
</tr>
<tr>
<td>PurchaseOrderRealTimeExport</td>
<td>Exports purchase orders to SAP ERP.</td>
</tr>
<tr>
<td>PurchaseOrgPull</td>
<td>Imports a set of purchasing organizations, by reading from SAP ERP.</td>
</tr>
<tr>
<td>ReceiptErrorImport (File Channel)</td>
<td>Imports the receipt error information from SAP ERP.</td>
</tr>
<tr>
<td>ReceiptExport (File Channel)</td>
<td>Exports the receipt data from SAP ERP.</td>
</tr>
<tr>
<td>ReceiptNumberImport (File Channel)</td>
<td>Imports the receipt data from SAP ERP.</td>
</tr>
<tr>
<td>ReceiptRealTimeExport</td>
<td>Exports receipts to SAP ERP.</td>
</tr>
<tr>
<td>ReleaseAuthorityPull</td>
<td>Imports release authorization information from SAP ERP.</td>
</tr>
<tr>
<td>RemittanceImport (File Channel)</td>
<td>Imports remittance data from SAP ERP to Ariba Procurement Solution using the File Channel.</td>
</tr>
<tr>
<td>RemittanceImport</td>
<td>Imports remittance data from SAP ERP to Ariba Procurement Solution.</td>
</tr>
<tr>
<td>TaxCodeLanguagePull</td>
<td>Defines translations for the names of tax codes.</td>
</tr>
<tr>
<td>TaxCodePull</td>
<td>Defines tax codes required in SAP ERP partitions and used only in SAP ERP partitions.</td>
</tr>
<tr>
<td>WBSElementPull</td>
<td>Defines the SAP ERP accounting entity WBSElement. The ProjectNumber field was not available in this event. As a result, WBSElements were not getting linked to the relevant ProjectNumbers in Ariba Procurement Solution. To ensure that the ProjectNumber field is included in this event, you must download SAP transport, U47K900070 from connect.ariba.com and import it into your SAP ERP system.</td>
</tr>
<tr>
<td>Import Supplier Data (Consolidated File)</td>
<td>Imports the supplier information from the SAP ERP system to the Ariba Procurement Solution system.</td>
</tr>
<tr>
<td>Import Supplier Location Data (Consolidated File)</td>
<td>Imports the supplier location for the supplier from the SAP ERP system to the Ariba Procurement Solution system.</td>
</tr>
<tr>
<td>Import Payment Terms Data (Consolidated File)</td>
<td>Imports the payment terms data from the SAP ERP system to the Ariba Procurement Solution system.</td>
</tr>
<tr>
<td>Event Name</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Import Remittance Location Data (Consolidated File)</td>
<td>Imports remittance location from SAP ERP system to the Ariba Procurement Solution system.</td>
</tr>
<tr>
<td>Import User Data (Consolidated File)</td>
<td>Imports user information from the SAP ERP system to the Ariba Procurement Solution system.</td>
</tr>
<tr>
<td>Import User to Group Mapping Data (Consolidated File)</td>
<td>Imports user groups from the SAP ERP system to the Ariba Procurement Solution system.</td>
</tr>
<tr>
<td>Import Purchase Organization Supplier Combo data (Consolidated File)</td>
<td>Imports the combinations of purchasing organizations and supplier information from the SAP ERP system to the Ariba Procurement Solution system.</td>
</tr>
<tr>
<td>Import Contracts in 5 CSV files</td>
<td>Imports contract information from the SAP ERP system to the Ariba Procurement Solution using CSV files.</td>
</tr>
<tr>
<td>Export Advance Payments</td>
<td>Exports approved advance payments from the Ariba Procurement Solution for integration with SAP ERP.</td>
</tr>
<tr>
<td>Export Payment Requests with Advance Payments</td>
<td>Exports payment information along with the advance payment information from the Ariba Procurement Solution for integration with SAP ERP.</td>
</tr>
<tr>
<td>Export Cancel Advance Payments</td>
<td>Exports advance payments with the status Canceling from the Ariba Procurement Solution for integration with SAP ERP.</td>
</tr>
<tr>
<td>Export Advance Payment Cancellation Reasons</td>
<td>Exports IDs and descriptions associated with the reasons for cancellation of the advance payments from the Ariba Procurement Solution to SAP ERP.</td>
</tr>
<tr>
<td>Import Advance Payment Remittance (File Channel)</td>
<td>Imports remittance information for advance payment document from SAP ERP to the Ariba Procurement Solution.</td>
</tr>
<tr>
<td>Import Advance Payment ID</td>
<td>Imports IDs returned by SAP ERP after successful upload of advance payments from the Ariba Procurement Solution.</td>
</tr>
<tr>
<td>Import Advance Payment Errors</td>
<td>Imports errors generated by SAP ERP while uploading advance payment documents sent by the Ariba Procurement Solution.</td>
</tr>
<tr>
<td>Import Cancel Advance Payment ID</td>
<td>Imports IDs of canceled advance payments returned by SAP ERP after successful upload of canceled advance payments from Ariba Procurement Solution.</td>
</tr>
<tr>
<td>Import Document Types for Advance Payment (File Channel)</td>
<td>Imports ERP document types that are used for advance payments from SAP ERP.</td>
</tr>
<tr>
<td>Import Translations for Advance Payment Document Types (File Channel)</td>
<td>Imports translations for ERP document types that are used for advance payments from SAP ERP to the Ariba Procurement Solution.</td>
</tr>
<tr>
<td>Import GL Indicators for Advance Payment (File Channel)</td>
<td>Imports ERP GL indicators that are used for advance payments from SAP ERP to the Ariba Procurement Solution.</td>
</tr>
<tr>
<td>Import Translations for Advance Payment GL Indicators (File Channel)</td>
<td>Imports translations for ERP GL indicators that are used for advance payments from SAP ERP to the Ariba Procurement Solution.</td>
</tr>
<tr>
<td>Export ERP Document Types for Advance Payment (File Channel)</td>
<td>Exports the ERP document types that are used for advance payments from SAP ERP to the Ariba Procurement Solution.</td>
</tr>
<tr>
<td>Export GL Indicators for Advance Payments (File Channel)</td>
<td>Exports the ERP GL indicators that are used for advance payments from SAP ERP to the Ariba Procurement Solution.</td>
</tr>
<tr>
<td>Event Name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Export Purchase Orders</td>
<td>The existing Export Purchase Orders web services task has been updated to include details about service items.</td>
</tr>
<tr>
<td>Export Service Sheets to External System</td>
<td>Exports service sheets from Ariba Procurement Solution to SAP ERP for approval.</td>
</tr>
<tr>
<td>Import Service Sheet Approval Status</td>
<td>Imports the approval statuses of service sheets from SAP ERP to the Ariba Procurement Solution.</td>
</tr>
</tbody>
</table>
Ariba SAP Objects

This section describes the SAP development objects provided by Ariba as part of the SAP integration and the naming conventions for those objects. The audience for this appendix includes:

- BASIS administrators who import Ariba’s SAP transports and need to see the list of objects they’re importing.
- ABAP developers who are responsible for customizing the Ariba SAP objects.

Naming Conventions for Ariba SAP Objects

The general naming conventions for Ariba SAP objects are as follows:

- **Package** names begin with the prefix `/ARBA/`.
- **Function group** names begin with the `/ARBA/` prefixes.
- **Function module** names begin with the prefix `/ARBA/`. These modules are customizable. Ensure that you do not modify the Ariba function modules.
- **Table** names begin with the prefix `/ARBA/`. Do not modify any of these Ariba core tables.
- **Structure** names begin with the prefix `/ARBA/`. The structures are core Ariba structures that you must not modify; however you can customize them.
Data element names begin with the prefix /ARBA/.

Message class: Ariba’s message class is named /ARBA/.

Summary of Ariba SAP packages

This section lists the packages and the Ariba SAP objects in those classes. The following table describes the packages:

<table>
<thead>
<tr>
<th>Package</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/MASTER_DATA_EXPORT</td>
<td>Ariba development objects</td>
</tr>
<tr>
<td>/ARBA/INVOICE_CREATE</td>
<td>Ariba on-demand invoice objects</td>
</tr>
<tr>
<td>/ARBA/REMITTANCE_FILE</td>
<td>Ariba on-demand remittance pull objects</td>
</tr>
<tr>
<td>/ARBA/PO_IMPORT</td>
<td>Ariba purchase order push object</td>
</tr>
<tr>
<td>/ARBA/CHANGEORDER</td>
<td>Ariba change purchase order PO push object</td>
</tr>
<tr>
<td>/ARBA/GOODS_RECEIPTS_CREATE</td>
<td>Ariba receipt push object</td>
</tr>
<tr>
<td>/ARBA/FILE_CHANNEL</td>
<td>Ariba file channel objects</td>
</tr>
<tr>
<td>/ARBA/INVOICE_INTEGRATION</td>
<td>Ariba Invoice object</td>
</tr>
<tr>
<td>/ARBA/EXPENSE_INTEGRATION</td>
<td>Ariba expense report object</td>
</tr>
<tr>
<td>/ARBA/GR_INTEGRATION</td>
<td>Ariba goods receipt object using File Channel</td>
</tr>
<tr>
<td>/ARBA/EXPENSE_REPORT</td>
<td>Ariba expense report object using File Channel</td>
</tr>
<tr>
<td>/ARBA/REMITTANCE_EXPORT</td>
<td>Ariba remittance export object using Web services</td>
</tr>
<tr>
<td>/ARBA/REMITTANCE_FILE</td>
<td>Ariba remittance file object using File Channel</td>
</tr>
<tr>
<td>/ARBA/UTIL</td>
<td>Ariba common utility</td>
</tr>
</tbody>
</table>

Each package topic describes its objects, including any of the following:

- Function groups and function modules
- Tables
- Structures
- Data elements
- Includes
- Message classes

/ARBA/MASTER_DATA_EXPORT Package

The /ARBA/MASTER_DATA_EXPORT package contains Ariba development objects for master data.
## Function Groups and Modules

The following are the function groups in the `/ARBA/MASTER_DATA_EXPORT` package:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/ARBA/MATERIAL_MANAGE_RFC</code></td>
<td>Ariba Materials Management RFCs</td>
</tr>
<tr>
<td><code>/ARBA/FICO_RFC</code></td>
<td>Ariba FICO RFCs</td>
</tr>
<tr>
<td><code>/ARBA/CORE_RFC</code></td>
<td>Ariba Core RFCs</td>
</tr>
<tr>
<td><code>/ARBA/IMG_RFC</code></td>
<td>Ariba IMG RFCs</td>
</tr>
<tr>
<td><code>/ARBA/TMG</code></td>
<td>/ARBA/TVARV maintenance dialog-generated</td>
</tr>
<tr>
<td><code>/ARBA/READ_CSV</code></td>
<td>Reads the CSV contents</td>
</tr>
<tr>
<td><code>/ARBA/MASTER_LANGUAGE_RFC</code></td>
<td>Ariba master data language pulls</td>
</tr>
<tr>
<td><code>/ARBA/DOWNLOAD_FILE</code></td>
<td>Ariba on-demand download CSV files</td>
</tr>
<tr>
<td><code>/ARBA/AUTH_PARAM</code></td>
<td>Ariba Direct Connectivity parameters</td>
</tr>
<tr>
<td><code>/ARBA/CHANGE_DOCS</code></td>
<td>Reads the change events for the incremental change transaction</td>
</tr>
<tr>
<td><code>/ARBA/DIRECT_CONNECTIVITY</code></td>
<td>Applicable to Direct Connectivity</td>
</tr>
<tr>
<td><code>/ARBA/SYSTID_MAP</code></td>
<td>Maps the system ID for the vendors</td>
</tr>
<tr>
<td><code>/ARBA/USER_RFC</code></td>
<td>Information on the users and user groups</td>
</tr>
</tbody>
</table>

In the `/ARBA/MASTER_DATA_EXPORT` package, the function group `/ARBA/TMG` contains only function modules automatically generated by the SAP maintenance dialog. The function modules in these function groups are not customizable and are not described in this document. This is applicable for the following function modules:

- `/ARBA/AUTH_PARAM`
- `/ARBA/CHANGE_DOCS`
- `/ARBA/SYSTID_MAP`

You cannot customize the `/ARBA/DIRECT_CONNECTIVITY` function module.

The function modules for the rest of the function groups are described in the following tables.

### Table 1: `/ARBA/MATERIAL_MANAGE_RFC` Function Group

<table>
<thead>
<tr>
<th>Function Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/ARBA/ACCOUNT_CATEGORY_EXPORT</code></td>
<td>The function module that exports the account assignment categories and their properties.</td>
</tr>
<tr>
<td><code>/ARBA/ACCOUNT_FIELD_EXPORT</code></td>
<td>The function module that exports the account category and field status information.</td>
</tr>
<tr>
<td><code>/ARBA/COMPANY_CODE_EXPORT</code></td>
<td>The function module that exports the company code information.</td>
</tr>
<tr>
<td><code>/ARBA/MATERIAL_GROUP_EXPORT</code></td>
<td>The function module that exports the material group information.</td>
</tr>
<tr>
<td><code>/ARBA/PLANT_EXPORT</code></td>
<td>The function module that exports the plant (ship To) information.</td>
</tr>
<tr>
<td><code>/ARBA/PLANT_PURCHASE_EXPORT</code></td>
<td>The function module that exports the plant and purchase organization information.</td>
</tr>
<tr>
<td>Function Module</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>/ARBA/PURCHASE_GROUP_EXPORT</td>
<td>The function module that exports the purchase group information.</td>
</tr>
<tr>
<td>/ARBA/PURCHASE_ORG_EXPORT</td>
<td>The function module that exports the purchase organization information.</td>
</tr>
<tr>
<td>/ARBA/VENDOR_EXPORT</td>
<td>The function module that exports the vendor and purchase organization information.</td>
</tr>
<tr>
<td>/ARBA/VENDOR_ONLY_EXPORT</td>
<td>The function module that exports the vendor information.</td>
</tr>
<tr>
<td>/ARBA/REMITTANCE_LOC_EXPORT</td>
<td>The function module that exports the remittance location information.</td>
</tr>
<tr>
<td>/ARBA/VENDOR_INCRMENT_EXPORT</td>
<td>The function module that exports the changed vendor information.</td>
</tr>
<tr>
<td>/ARBA/SUPPLIER_LOCATION_EXPORT</td>
<td>The function module that exports the supplier location information.</td>
</tr>
</tbody>
</table>

Table 12: /ARBA/FICO_RFC Function Group

<table>
<thead>
<tr>
<th>Function Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/ASSET_EXPORT</td>
<td>The function module that exports the asset information.</td>
</tr>
<tr>
<td>/ARBA/COST_CENTER_EXPORT</td>
<td>The function module that exports the cost center information.</td>
</tr>
<tr>
<td>/ARBA/GENERAL_LEDGER_EXPORT</td>
<td>The function module that exports the general ledger information.</td>
</tr>
<tr>
<td>/ARBA/INTERNAL_ORDER_EXPORT</td>
<td>The function module that exports the internal order information.</td>
</tr>
<tr>
<td>/ARBA/WBS_EXPORT</td>
<td>The function module that exports the WBS information.</td>
</tr>
</tbody>
</table>

Table 13: /ARBA/CORE_RFC Function Group

<table>
<thead>
<tr>
<th>Function Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/LAST_CHANGE_DATE</td>
<td>Gets last change date for an object given its object class</td>
</tr>
<tr>
<td>/ARBA/PARAMETER</td>
<td>Gets the value of a parameter</td>
</tr>
<tr>
<td>/ARBA/PREFILTER</td>
<td>Retrieves prefiltering restrictions</td>
</tr>
<tr>
<td>/ARBA/SELECT_LIST</td>
<td>Returns fields of structure</td>
</tr>
</tbody>
</table>

Table 14: /ARBA/IMG_RFC Function Group

<table>
<thead>
<tr>
<th>Function Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CURRENCY_CONVERSION_EXPORT</td>
<td>Ariba Adapter Currency Conversion Pull Function Module</td>
</tr>
<tr>
<td>/ARBA/TAX_CODE_EXPORT</td>
<td>Tax Code Pull for Use in Ariba</td>
</tr>
</tbody>
</table>

Table 15: /ARBA/DOWNLOAD_FILE Function Group

<table>
<thead>
<tr>
<th>Function Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CONVERT_TO_CSV_FORMAT</td>
<td>Converts SAP data into CSV format</td>
</tr>
<tr>
<td>/ARBA/CONVERT_TO_TEXT_FORMAT</td>
<td>Converts the data into text format</td>
</tr>
<tr>
<td>/ARBA/DATA_CONVERT_WRITE_FILE</td>
<td>Converts the SAP data and downloads the data</td>
</tr>
</tbody>
</table>

Table 16: /ARBA/READ_CSV Function group

<table>
<thead>
<tr>
<th>Function Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/KCD_CSV_FILE_CONVERSION</td>
<td>Parse the data in CSV format</td>
</tr>
</tbody>
</table>
Table 17: /ARBA/MASTER_LANGUAGE_RFC Function group

<table>
<thead>
<tr>
<th>Function Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/ACCOUNT_CAT_NAMES_EXPORT</td>
<td>Pulls account assignment category multi-lingual names</td>
</tr>
<tr>
<td>/ARBA/COST_CENTER_NAMES_EXPORT</td>
<td>Pulls cost center multi-lingual names</td>
</tr>
<tr>
<td>/ARBA/GENERAL_LDGR_NAMS_EXPORT</td>
<td>Pull general ledgers multi-lingual names</td>
</tr>
<tr>
<td>/ARBA/MATERIAL_GRP_NAME_EXPORT</td>
<td>Pulls material group multi-lingual names</td>
</tr>
<tr>
<td>/ARBA/TAX_CODE_NAMES_EXPORT</td>
<td>Pulls tax code multi-lingual names</td>
</tr>
<tr>
<td>/ARBA/PAYMENTTERM_NAME_EXPORT</td>
<td>Pulls payment terms</td>
</tr>
</tbody>
</table>

Tables

The following are the tables for the /ARBA/MASTER_DATA_EXPORT package:

Table 18: /ARBA/MASTER_DATA_EXPORT Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/TVARV</td>
<td>TVARV generalized for fields</td>
</tr>
<tr>
<td>/ARBA/AUTH_PARAM</td>
<td>Ariba Direct Connectivity parameters</td>
</tr>
<tr>
<td>/ARBA/FIELD_MAP</td>
<td>Maps the SAP and Ariba procurement solution system field values</td>
</tr>
<tr>
<td>/ARBA/INCR_DTTIM</td>
<td>Stores the time stamp for the last run master data export tasks</td>
</tr>
<tr>
<td>/ARBA/SYSTID_MAP</td>
<td>Maps the system ID for the vendors</td>
</tr>
<tr>
<td>/ARBA/USR_GRP</td>
<td>Stores the Ariba user roles</td>
</tr>
</tbody>
</table>

Structures

The following are the structures for the /ARBA/MASTER_DATA_EXPORT package:

Table 19: /ARBA/MASTER_DATA_EXPORT Structures

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/ACCOUNT_ASSIGNMENT_CATGY</td>
<td>Account assignment category and its properties</td>
</tr>
<tr>
<td>/ARBA/ACCOUNT_CATEGORY</td>
<td>Account Category/Field Status Group combination</td>
</tr>
<tr>
<td>/ARBA/ACCOUNT_CATEGORY_NAMES</td>
<td>Names for account categories</td>
</tr>
<tr>
<td>/ARBA/ASSET</td>
<td>Stores main asset information for use in Ariba</td>
</tr>
<tr>
<td>/ARBA/BP_LANG_STR</td>
<td>Budget Period Language Pull</td>
</tr>
<tr>
<td>/ARBA/BP_STR</td>
<td>Budget Period Sturcture</td>
</tr>
<tr>
<td>/ARBA/CELLS</td>
<td>Ariba cells in flat structure for flexible excel upload.</td>
</tr>
<tr>
<td>Structure</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>/ARBA/COMMIT_LANG_STR</td>
<td>Commitment Item language pull csv structure</td>
</tr>
<tr>
<td>/ARBA/COMPANY_CODE</td>
<td>Company code, name, and other details</td>
</tr>
<tr>
<td>/ARBA/COST_CENTER</td>
<td>To export all cost structure information for use in Ariba</td>
</tr>
<tr>
<td>/ARBA/COST_CENTER_NAMES</td>
<td>Cost center names</td>
</tr>
<tr>
<td>/ARBA/CURRENCY_CONVERSION</td>
<td>Currency Conversion Type information for use in Ariba</td>
</tr>
<tr>
<td>/ARBA/CURRENCY_TIME_ZONE</td>
<td>Structure for currency conversion type information for Ariba AribaFCURR F.</td>
</tr>
<tr>
<td>/ARBA/FM01T_STR</td>
<td>FMArea language structure</td>
</tr>
<tr>
<td>/ARBA/FMAREA_LANG_STR</td>
<td>FMArea language pull csv structure</td>
</tr>
<tr>
<td>/ARBA/FMBUDGETPD_STR</td>
<td>Budget Period Query Structure</td>
</tr>
<tr>
<td>/ARBA/FMBUDGETPDT_STR</td>
<td>Budget Period Language Pull</td>
</tr>
<tr>
<td>/ARBA/FMCIT_STR</td>
<td>Commitment item language structure</td>
</tr>
<tr>
<td>/ARBA/FMFCTR_STR</td>
<td>Funds Center</td>
</tr>
<tr>
<td>/ARBA/FMFCTR7_STR</td>
<td>Fund Center Language Pull</td>
</tr>
<tr>
<td>/ARBA/FMFINCODE_STR</td>
<td>FiFM: Financing code</td>
</tr>
<tr>
<td>/ARBA/FMFINT_STR</td>
<td>Fund Language Pull</td>
</tr>
<tr>
<td>/ARBA/FMGRANT_LANG_STR</td>
<td>Grant Language Pull</td>
</tr>
<tr>
<td>/ARBA/FMGRANT_STR</td>
<td>FM Grant Structure</td>
</tr>
<tr>
<td>/ARBA/FUNCTAREA_LANG_STR</td>
<td>Functional Area Language Pull</td>
</tr>
<tr>
<td>/ARBA/FUND_LANG_STR</td>
<td>Fund language pull</td>
</tr>
<tr>
<td>/ARBA/FUND_STR</td>
<td>Fund Export</td>
</tr>
<tr>
<td>/ARBA/FUNDCTR_LANG_STR</td>
<td>Fund Center language Pull</td>
</tr>
<tr>
<td>/ARBA/FUNDCTR_STR</td>
<td>Funds Center Export</td>
</tr>
<tr>
<td>/ARBA/GENERAL_LEDGER</td>
<td>General ledger information</td>
</tr>
<tr>
<td>/ARBA/GENERAL_LEDGER_NAMES</td>
<td>Names of general ledgers</td>
</tr>
<tr>
<td>/ARBA/GMGR_STR</td>
<td>Grant Query Structure</td>
</tr>
<tr>
<td>/ARBA/GMGRTEXTS_STR</td>
<td>Grant Language Pull</td>
</tr>
<tr>
<td>/ARBA/MATERIAL_GROUP</td>
<td>To export all material group information for use in Ariba</td>
</tr>
<tr>
<td>/ARBA/MATERIAL_GROUP_NAMES</td>
<td>Material group names</td>
</tr>
<tr>
<td>/ARBA/ORDER</td>
<td>Holds order numbers for Ariba</td>
</tr>
<tr>
<td>/ARBA/PAYMENT_TERMS</td>
<td>Payment terms</td>
</tr>
<tr>
<td>/ARBA/PAYMENT_TERM_NAMES</td>
<td>Payment terms names</td>
</tr>
<tr>
<td>Structure</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>/ARBA/PLANT</td>
<td>To export all plant information for use in Ariba</td>
</tr>
<tr>
<td>/ARBA/PLANT_PURCHASE</td>
<td>Plant/Purchase Orgs combinations for use in Ariba</td>
</tr>
<tr>
<td>/ARBA/PURCHASE_ORGANISATION</td>
<td>To export purchase organizations for use in Ariba</td>
</tr>
<tr>
<td>/ARBA/PURCHASING_GROUP</td>
<td>To export all purchasing Groups for use in Ariba</td>
</tr>
<tr>
<td>/ARBA/TAX_CODE</td>
<td>Tax code information for use in Ariba</td>
</tr>
<tr>
<td>/ARBA/TAX_CODE_NAMES</td>
<td>Tax code names for use in Ariba</td>
</tr>
<tr>
<td>/ARBA/TFKBT_STR</td>
<td>Functional Area Language Pull</td>
</tr>
<tr>
<td>/ARBA/USER</td>
<td>User information</td>
</tr>
<tr>
<td>/ARBA/USER_GROUP</td>
<td>User group names</td>
</tr>
<tr>
<td>/ARBA/USER_HR</td>
<td>User HR information</td>
</tr>
<tr>
<td>• /ARBA/VENDOR</td>
<td>To export vendor information for use in Ariba</td>
</tr>
<tr>
<td>• /ARBA/SUPPLIER_LOCATION</td>
<td></td>
</tr>
<tr>
<td>• /ARBA/REMITTANCE_LOCATION</td>
<td></td>
</tr>
<tr>
<td>/ARBA/VENDOR_PO</td>
<td>To export vendor/purchase org information for use in Ariba</td>
</tr>
<tr>
<td>/ARBA/WBS</td>
<td>WBS element information</td>
</tr>
<tr>
<td>/ARBA/WTAB</td>
<td>Table structure for “WHERE (itab)” statement</td>
</tr>
</tbody>
</table>

**Proxy Structures**

The following are the proxy structures for the `/ARBA/MASTER_DATA_EXPORT` package:

`/ARBA/DT_MASTER_DATA`

- `/ARBA/DT_MASTER_DATA_ATTACHMENT`
- `/ARBA/DT_MASTER_DATA_HEADER`
- `/ARBA/DT_MASTER_DATA_PARAMETER`
- `/ARBA/DT_MASTER_DATA_PARAM_TAB`
- `/ARBA/DT_MASTER_DATA_REQ`
- `/ARBA/DT_MASTER_DATA_REQ_ATTACHMENT`
- `/ARBA/DT_MASTER_DATA_REQ_HEADER`
- `/ARBA/DT_MASTER_DATA_REQ_PARAMETER`
- `/ARBA/DT_MASTER_DATA_REQ_PARAM_TAB`
- `/ARBA/DT_MASTER_DATA_REQ_P_TAB`
- `/ARBA/DT_MASTER_DATA_RES`
- `/ARBA/DT_RESPONSE`
- `/ARBA/DT_MASTER_DATA_PARAM_TAB`
- `/ARBA/DT_MASTER_DATA_REQ_P_TAB`
# Table Types

The following are the table types for the structures in the `/ARBA/MASTER_DATA_EXPORT` package:

<table>
<thead>
<tr>
<th>Table Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/ARBA/ACCOUNT_ASSIGNMENT_CTG_T</code></td>
<td>Table type for Account Assignment Category</td>
</tr>
<tr>
<td><code>/ARBA/ACCOUNT_CATEGORY_NAMES_T</code></td>
<td>Table type for names of the Account Category/Field Status Group combination</td>
</tr>
<tr>
<td><code>/ARBA/ACCOUNT_CATEGORY_T</code></td>
<td>Table type for Account Category/Field Status Group combination</td>
</tr>
<tr>
<td><code>/ARBA/ACCOUNT_FIELD_T</code></td>
<td>Table type for Account Category/Field Status</td>
</tr>
<tr>
<td><code>/ARBA/ADDFAX_T</code></td>
<td>BAPI structure Fax Numbers</td>
</tr>
<tr>
<td><code>/ARBA/ADDPAG_T</code></td>
<td>BAPI structure Pager Numbers</td>
</tr>
<tr>
<td><code>/ARBA/ADDSMTP_T</code></td>
<td>E-mail addresses BAPI Structure</td>
</tr>
<tr>
<td><code>/ARBA/ADDDTEL_T</code></td>
<td>BAPI structure Telephone Numbers</td>
</tr>
<tr>
<td><code>/ARBA/ADDTLX_T</code></td>
<td>BAPI structure Telex Numbers</td>
</tr>
<tr>
<td><code>/ARBA/ADDTTX_T</code></td>
<td>BAPI structure Teletex Numbers</td>
</tr>
<tr>
<td><code>/ARBA/ASSET_T</code></td>
<td>Table type for Asset information for use in Ariba</td>
</tr>
<tr>
<td><code>/ARBA/BAPIRET_DC</code></td>
<td>Table type for Return Parameter</td>
</tr>
<tr>
<td><code>/ARBA/BP_LANG_STR_T</code></td>
<td>Budget Period Language Pull</td>
</tr>
<tr>
<td><code>/ARBA/BP_STR_T</code></td>
<td>Budget Period Table Type</td>
</tr>
<tr>
<td><code>/ARBA/CHANGE_DOC_T</code></td>
<td>Table type for Change Docs</td>
</tr>
<tr>
<td><code>/ARBA/COMMUT_LANG_STR_T</code></td>
<td>Commitment item language pull CSV structure table</td>
</tr>
<tr>
<td><code>/ARBA/COMPANY_CODE_T</code></td>
<td>Table type for Company Code, Name, and other details</td>
</tr>
<tr>
<td><code>/ARBA/COST_CENTER_NAMES_T</code></td>
<td>Table type for Cost Center names</td>
</tr>
<tr>
<td><code>/ARBA/COST_CENTER_T</code></td>
<td>Table type for Cost Centers</td>
</tr>
<tr>
<td><code>/ARBA/CURRENCY_CONVERSION_T</code></td>
<td>Table type for Currency Conversion Type information for use in Ariba</td>
</tr>
<tr>
<td><code>/ARBA/EXCHANGE_LOG_DATA_TAB</code></td>
<td>Proxy table type (generated)</td>
</tr>
<tr>
<td><code>/ARBA/FM01T_STR_T</code></td>
<td>FMArea Language Table Type</td>
</tr>
<tr>
<td><code>/ARBA/FMAREA_LANG_STR_T</code></td>
<td>FMArea Language pull</td>
</tr>
<tr>
<td><code>/ARBA/FMGRANT_LANG_STR_T</code></td>
<td>Grant Language Pull</td>
</tr>
<tr>
<td><code>/ARBA/FMGRANT_STR_T</code></td>
<td>FM Table Type</td>
</tr>
<tr>
<td><code>/ARBA/FUNCTAREA_LANG_STR_T</code></td>
<td>Functional Area Language Pull</td>
</tr>
<tr>
<td><code>/ARBA/FUNDCTR_STR_T</code></td>
<td>Funds Center Export</td>
</tr>
<tr>
<td><code>/ARBA/FUNDCTR_LANG_STR_T</code></td>
<td>Fund Center Language Pull</td>
</tr>
<tr>
<td><code>/ARBA/FUND_LANG_STR_T</code></td>
<td>Fund Language Pull</td>
</tr>
<tr>
<td><code>/ARBA/FUND_STR_T</code></td>
<td>Fund Export</td>
</tr>
</tbody>
</table>
## Table Type

<table>
<thead>
<tr>
<th>Table Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/GENERAL_LEDGER_NAMES_T</td>
<td>Table type for General Ledger names</td>
</tr>
<tr>
<td>/ARBA/GENERAL_LEDGER_T</td>
<td>Table type for General Ledgers</td>
</tr>
<tr>
<td>/ARBA/INTERNAL_ORDER_T</td>
<td>Table type For Internal Orders</td>
</tr>
<tr>
<td>/ARBA/MATERIAL_GROUP_NAMES_T</td>
<td>Table type For Material Group Names</td>
</tr>
<tr>
<td>/ARBA/MATERIAL_GROUP_NAME_T</td>
<td>Table type for Material Group names</td>
</tr>
<tr>
<td>/ARBA/MATERIAL_GROUP_T</td>
<td>Table type for the Material Groups</td>
</tr>
<tr>
<td>/ARBA/PARAMETER1_T</td>
<td>Replaces parameter (Length 18 -&gt; 40)</td>
</tr>
<tr>
<td>/ARBA/PARAMETER_T</td>
<td>Table with User Parameters</td>
</tr>
<tr>
<td>/ARBA/PAYMENT_TERMS_T</td>
<td>Table type for Payment terms</td>
</tr>
<tr>
<td>/ARBA/PAYMENT_TERM_NAMES_T</td>
<td>Table type for Payment terms names</td>
</tr>
<tr>
<td>/ARBA/PLANT_PURCHASE_T</td>
<td>Table type for Plant/Purchase Orgs Combinations information</td>
</tr>
<tr>
<td>/ARBA/PLANT_T</td>
<td>Table type for Plant information</td>
</tr>
<tr>
<td>/ARBA/PREFILTER_T</td>
<td>Table type For Prefilter</td>
</tr>
<tr>
<td>/ARBA/PURCHASE_ORGANISATION_T</td>
<td>Table type for Purchasing organizations</td>
</tr>
<tr>
<td>/ARBA/PURCHASING_GROUP_T</td>
<td>Table type for Purchasing groups</td>
</tr>
<tr>
<td>/ARBA/REMITTANCE_LOCATION_T</td>
<td>Table type for Remittance locations</td>
</tr>
<tr>
<td>/ARBA/SDIT_QRY_T</td>
<td>SG01, Internal Structure: Input Table for Dynamic SQL</td>
</tr>
<tr>
<td>/ARBA/SUPPLIER_LOCATION_T</td>
<td>Table type for Supplier locations</td>
</tr>
<tr>
<td>/ARBA/TAX_CODE_NAMES_T</td>
<td>Table type for Tax code names</td>
</tr>
<tr>
<td>/ARBA/TAX_CODE_T</td>
<td>Table type for Tax codes</td>
</tr>
<tr>
<td>/ARBA/USER_GROUP_T</td>
<td>Table type for User groups</td>
</tr>
<tr>
<td>/ARBA/USER_T</td>
<td>Table type for Users</td>
</tr>
<tr>
<td>/ARBA/VENDOR_PO_T</td>
<td>Table type for Vendor/Purchase Org</td>
</tr>
<tr>
<td>/ARBA/VENDOR_T</td>
<td>Table type for Vendors</td>
</tr>
<tr>
<td>/ARBA/WBS_T</td>
<td>Table type for WBS elements</td>
</tr>
</tbody>
</table>

## Data Elements

The following are the data elements for the /ARBA/MASTER_DATA_EXPORT package:

<table>
<thead>
<tr>
<th>Data Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/FM_BUDGET_PERIOD_NAME</td>
<td>Budget Period Name</td>
</tr>
<tr>
<td>/ARBA/FNAME_ARIBA</td>
<td>Field name in Ariba</td>
</tr>
<tr>
<td>Data Element</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>/ARBA/FNAME_SAP</td>
<td>Field name for SAP</td>
</tr>
<tr>
<td>/ARBA/LOWER_CASE</td>
<td>Field name case sensitive</td>
</tr>
<tr>
<td>/ARBA/PASSWORD</td>
<td>Field name for password</td>
</tr>
<tr>
<td>/ARBA/STRUCTURE</td>
<td>Field name for structure</td>
</tr>
</tbody>
</table>

### Message Class

The following is the message class for the /ARBA/MASTER_DATA_EXPORT package:

<table>
<thead>
<tr>
<th>Message Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/MESSAGE_CLASS</td>
<td>Error messages for Ariba SAP ABAP code</td>
</tr>
</tbody>
</table>

### BAdI

The following is the BAdI information for the /ARBA/MASTER_DATA_EXPORT package:

<table>
<thead>
<tr>
<th>BAdI Class</th>
<th>BAdI Interface</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_EX_MASTER_DATA</td>
<td>/ARBA/IF_EXP_MASTER_DATA</td>
<td>/ARBA/MASTER_DATA</td>
<td>BADI definition For Master Data Pull</td>
</tr>
</tbody>
</table>

### /ARBA/PURCHREQP2P Package

This package contains the Ariba requisition objects.

### /ARBA/PURCHREQP2P Tables

The following is the table for the /ARBA/PURCHREQP2P package:

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/PR_HISTORY</td>
<td>SAP requisitions which were pushed to SAP by Ariba</td>
</tr>
</tbody>
</table>
/ARBA/PURCHREQP2P Table Types

The following are the table types for the /ARBA/PURCHREQP2P:

<table>
<thead>
<tr>
<th>Proxy Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BNFPO</td>
<td>Requisition line item</td>
</tr>
<tr>
<td>/ARBA/TY_BAPIMEREQACCOUNT</td>
<td>Table type for BAPIMEREQACCOUNT</td>
</tr>
<tr>
<td>/ARBA/TY_BAPIMEREQITEMIMP</td>
<td>Table type for item map</td>
</tr>
</tbody>
</table>

/ARBA/PURCHREQP2P Structures

The following are the structures for the /ARBA/PURCHREQP2P package:

Table 21: /ARBA/PURCHREQP2P Structures

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BAPIACCRPO</td>
<td>For posting in FI/CO: commitments (currency fields)</td>
</tr>
<tr>
<td>/ARBA/BAPIACPR00</td>
<td>For posting in accounting: requisition (item)</td>
</tr>
<tr>
<td>/ARBA/BAPIMEREQITEMIMP</td>
<td>Change toolbar for requisition (item)</td>
</tr>
</tbody>
</table>

/ARBA/PURCHREQP2P Domain

The following is the domain in the /ARBA/PURCHREQP2P package:

<table>
<thead>
<tr>
<th>Domain</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/REQ_FILTER</td>
<td>BADI filter parameter for the requisition transaction</td>
</tr>
</tbody>
</table>

/ARBA/PURCHREQP2P Function Group

The /ARBA/PURCHREQP2P package contains the following function group:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/PURREQ</td>
<td>Ariba Procurement Solution requisition integration</td>
</tr>
</tbody>
</table>
/ARBA/PURCHREQP2P Function Modules

The /ARBA/PURCHREQP2P package contains the following function modules:

<table>
<thead>
<tr>
<th>Function Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/PURREQ</td>
<td>Requisition function group</td>
</tr>
<tr>
<td>/ARBA/PURREQ_DELETE</td>
<td>Mark requisition line item for deletion</td>
</tr>
</tbody>
</table>

/ARBA/PURCHREQP2P BAdI

The following is the BAdI information for the /ARBA/PURCHREQP2P package:

<table>
<thead>
<tr>
<th>BAdI Class (ABAP Objects)</th>
<th>BAdI Interface (ABAP Objects)</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_PURREQ_P2P</td>
<td>/ARBA/IF_PURREQ</td>
<td>/ARBA/PURREQ</td>
<td>Utility class for budget check</td>
</tr>
</tbody>
</table>

/ARBA/INVOICE_CREATE Package

This package contains the Ariba invoice objects.

/ARBA/INVOICE_CREATE Tables

The following are the tables for the /ARBA/INVOICE_CREATE package:

Table 22: /ARBA/INVOICE_CREATE Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/PO_HISTORY</td>
<td>SAP POs which were pushed to SAP by Ariba</td>
</tr>
</tbody>
</table>

/ARBA/INVOICE_CREATE Structures

The following are the structures for the /ARBA/INVOICE_CREATE package:

Table 23: /ARBA/INVOICE_CREATE Structures

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/INVOICE_CSV_HEADER</td>
<td>Ariba CSV structure for invoice header</td>
</tr>
</tbody>
</table>
## /ARBA/INVOICE_CREATE Table Types

The following are the table types for the /ARBA/INVOICE_CREATE:

Table 24: /ARBA/INVOICE_CREATE -- Structures

<table>
<thead>
<tr>
<th>Proxy Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/INVOICE_CSV_LINE_T</td>
<td>Table type for invoice CSV line</td>
</tr>
<tr>
<td>/ARBA/INVOICE_CSV_HEADER_T</td>
<td>Table type for invoice CSV header</td>
</tr>
<tr>
<td>/ARBA/INVOICE_CSV_TAX_T</td>
<td>Table type for invoice CSV tax</td>
</tr>
<tr>
<td>/ARBA/INVOICE&gt;Create_ITEM_T</td>
<td>Table type for item data</td>
</tr>
<tr>
<td>/ARBA/INVOICE_CREATE_TAX_T</td>
<td>Table type for invoice create tax</td>
</tr>
<tr>
<td>/ARBA/INVOICE_CSV_ACCT_DATA_T</td>
<td>Table type CSV structure for invoice accounting data</td>
</tr>
<tr>
<td>/ARBA/INVOICE_CSV_ERROR_T</td>
<td>Table type for CSV error</td>
</tr>
<tr>
<td>/ARBA/INVOICE_CSV_NUMBER_PUL_T</td>
<td>Table type for Number Pull</td>
</tr>
</tbody>
</table>

## /ARBA/INVOICE_CREATE Programs

The following are the programs in the /ARBA/INVOICE_CREATE package:

Table 25: /ARBA/INVOICE_CREATE -- Programs

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CR_INVOICE_CREATE</td>
<td>Create Ariba invoices into SAP</td>
</tr>
<tr>
<td>/ARBA/CRINVOICE_READ</td>
<td>Include program for invoices</td>
</tr>
<tr>
<td>/ARBA/CRINVOICE_ERROR_HANDLING</td>
<td>Include program for invoices</td>
</tr>
</tbody>
</table>
BAdI

The following is the BAdI information for the /ARBA/INVOICE_CREATE package:

<table>
<thead>
<tr>
<th>BAdI Class (ABAP Objects)</th>
<th>BAdI Interface (ABAP Objects)</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_EX_INVOICE_LOAD</td>
<td>/ARBA/IF_EX_INVOICE_LOAD</td>
<td>/ARBA/INVOICE_LOAD</td>
<td>Add-in to load Ariba invoices into SAP</td>
</tr>
</tbody>
</table>

/ARBA/REMITTANCE_EXPORT Package

This class contains Ariba remittance pull programs.

/ARBA/REMITTANCE_EXPORT Function Group

The following is the function group in the /ARBA/REMITTANCE_EXPORT package:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/REMITTANCE_RFC</td>
<td>Function group for the Remittance RFC</td>
</tr>
</tbody>
</table>

/ARBA/REMITTANCE_EXPORT Data Element

The following is the element in the /ARBA/REMITTANCE_EXPORT package:

<table>
<thead>
<tr>
<th>Data Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/REM_DATETIME</td>
<td>Data element for date time stamp</td>
</tr>
</tbody>
</table>

/ARBA/REMITTANCE_EXPORT Function Group

The following is the function group in the /ARBA/REMITTANCE_EXPORT package:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/REMITTANCE_RFC</td>
<td>Function group for the Remittance RFC</td>
</tr>
</tbody>
</table>
/ARBA/REMITTANCE_EXPORT Structures

The following are the structures in the /ARBA/REMITTANCE_EXPORT package:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/REMDETAIL</td>
<td>Ariba payment export line items</td>
</tr>
<tr>
<td>/ARBA/REMHEADER</td>
<td>Ariba payment export line items</td>
</tr>
</tbody>
</table>

/ARBA/REMITTANCE_EXPORT Table Types

The following are the table types in the /ARBA/REMITTANCE_EXPORT package:

<table>
<thead>
<tr>
<th>Table Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/REMDETAIL_T</td>
<td>Table type for Ariba payment export line items</td>
</tr>
<tr>
<td>/ARBA/REMHEADER_T</td>
<td>Table type for Ariba payment export line items</td>
</tr>
</tbody>
</table>

BAdI

The following is the BAdI information for the /ARBA/REMITTANCE_EXPORT package:

<table>
<thead>
<tr>
<th>BAdI Class (ABAP Objects)</th>
<th>BAdI Interface (ABAP Objects)</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_EX_REMIT_EXPORT</td>
<td>/ARBA/IF_EX_REMIT_EXPORT</td>
<td>/ARBA/REMIT_EXPORT</td>
<td>BAdI for remittance export</td>
</tr>
</tbody>
</table>

/ARBA/PO_IMPORT Package

This package contains Ariba purchase order export objects.

/ARBA/PO_IMPORT Function Group

The following is the function group in the /ARBA/PO_IMPORT package:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/PO_RFC</td>
<td>Function group for PO RFC’s</td>
</tr>
</tbody>
</table>
### /ARBA/PO_IMPORT Table Types

The following are the table type structures in the /ARBA/PO_IMPORT package:

<table>
<thead>
<tr>
<th>Table Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BAPIESKLC_T</td>
<td>Table type for Acct Assgt Distr. for service lines</td>
</tr>
<tr>
<td>/ARBA/BAPIESLLC_T</td>
<td>Table type for Communication Structure: Create Service Line</td>
</tr>
<tr>
<td>/ARBA/BAPIESUCC_T</td>
<td>Table type for Communication Structure: Contract Limits</td>
</tr>
<tr>
<td>/ARBA/BAPIESUHC_T</td>
<td>Table type for Communication Structure: Limits</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOACCOUNTX_T</td>
<td>Table type for Account Assignment Fields in Purchase Order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOACCOUNT_T</td>
<td>Table type for Account Assignment Fields for Purchase Order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOADDRDELIVERY_T</td>
<td>Table type for Address Structure BAPIADDR1 for Inbound Delivery</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCONDHEADERX_T</td>
<td>Table type for conditions (Header, Change Bar)</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCONDHEADER_T</td>
<td>Table type for conditions</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCONDX_T</td>
<td>Table type for conditions in purchase order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCOND_T</td>
<td>Table type for conditions in purchase order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOITEM_T</td>
<td>Table type for purchase order item</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSCHEDULE_T</td>
<td>Table type for fields for purchase order delivery schedule</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSCHEDULX_T</td>
<td>Table type for fields for schedule lines in purchase order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOTEXT_T</td>
<td>Table type for texts: (Header or Item)</td>
</tr>
<tr>
<td>/ARBA/BAPIPAREX_T</td>
<td>Table type for BAPI parameter ExtensionIn/ExtensionOut</td>
</tr>
<tr>
<td>/ARBA/BAPIEIKPX_T</td>
<td>Foreign trade: Import/Export Header Data (Change Parameter)</td>
</tr>
<tr>
<td>/ARBA/BAPIEIKP_T</td>
<td>Foreign trade: Export/Import: Header Data</td>
</tr>
<tr>
<td>/ARBA/BAPIEIPOX_T</td>
<td>Table type for Import/Export: Item Data (Change Bar)</td>
</tr>
<tr>
<td>/ARBA/BAPIEIPO_T</td>
<td>Table type for foreign trade: Export/Import: Item Data</td>
</tr>
<tr>
<td>/ARBA/BAPIEKKOP_T</td>
<td>Table type for transfer structure for partner roles purchase order create</td>
</tr>
<tr>
<td>/ARBA/BAPIEKLCLC_T</td>
<td>Table type for: Acct Assgt Distr. for service lines</td>
</tr>
<tr>
<td>/ARBA/BAPIELLLC_T</td>
<td>Table type for create service lines</td>
</tr>
<tr>
<td>/ARBA/BAPIESLTX_T</td>
<td>Table type for BAPI Services long text</td>
</tr>
<tr>
<td>/ARBA/BAPIITEMSHIPX_T</td>
<td>BAPI Shipping Data Change parameter</td>
</tr>
<tr>
<td>/ARBA/BAPIITEMSHIP_T</td>
<td>BADI shipping data for stock transport orders</td>
</tr>
<tr>
<td>/ARBA/BAPIMEDCM_ALLVERSIONS_T</td>
<td>Table type for version management - all version data</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPACCOUNT_T</td>
<td>Table type for Account Assignment fields for purchase order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOACCOUNTPROFTS_T</td>
<td>Table type for BAPI_PROFITABILITY_SEGMENT</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOACCOUNTX_T</td>
<td>Table type for Account Assignment fields in purchase order</td>
</tr>
<tr>
<td>Table Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOADDDELIVERY_T</td>
<td>Table type for BAPIADDRI for Inbound Delivery</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCOMPONENTX_T</td>
<td>Update information for components in BUS2012 API</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCOMPONENT_T</td>
<td>BADI table for components</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCONDHEDERX_T</td>
<td>Table type for conditions (Header, Change Bar)</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCONDHEDER_T</td>
<td>Table type for conditions (Header)</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOITEM1_T</td>
<td>Structure for BAPItem</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOITMX_T</td>
<td>Table type for purchase order item data (Change Toolbar)</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOITEM_T</td>
<td>Table type for purchase order item</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSCEDUX_T</td>
<td>Table type for fields for schedule lines in purchase orders</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSCEDULE_T</td>
<td>Table type for fields for purchase order delivery schedule</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSHIPPEXP_T</td>
<td>Export structure for shipping data</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOTEXTXEDER_T</td>
<td>Table type for header texts</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOTXT_T</td>
<td>Table type for texts: (Header or Item)</td>
</tr>
<tr>
<td>/ARBA/BAPIMOCONDX_T</td>
<td>Table type for conditions in purchase orders</td>
</tr>
<tr>
<td>/ARBA/BAPIMOCOND_T</td>
<td>Table type for conditions in purchase orders</td>
</tr>
<tr>
<td>/ARBA/BAPIREX_T</td>
<td>Table type for BAPI parameter ExtensionIn/ExtensionOut</td>
</tr>
<tr>
<td>/ARBA/BAPISUCC_T</td>
<td>Table type for Communication Structure: Contract Limits</td>
</tr>
<tr>
<td>/ARBA/BAPISUHC_T</td>
<td>Table type for limits</td>
</tr>
<tr>
<td>/ARBA/ERROR_MESSAGE_T</td>
<td>Table type for error messages</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIEIPOX_T</td>
<td>Foreign Trade: Change Parameter: Item Data</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIEIPO_T</td>
<td>Foreign Trade: Item Data</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIEKKOP_T</td>
<td>Partner</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIESKLC_T</td>
<td>External Services: Account Assignment Distribution for Services</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIESLLC_T</td>
<td>External Services: Service Lines</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIESLLTX_T</td>
<td>External Services: Service Long Text</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIESUHC_T</td>
<td>External Services: Limits</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIESUCC_T</td>
<td>External Services: Contract Limits</td>
</tr>
<tr>
<td>/ARBA/IT_BAPITEMSHIPX_T</td>
<td>BAPI Shipping Data Change parameter</td>
</tr>
<tr>
<td>/ARBA/IT_BAPITEMSHIP_T</td>
<td>BAPI Shipping data for stock transport orders</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMED_ALLVERSIONS_T</td>
<td>All versions (Export Parameter)</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOACCOUNTX_T</td>
<td>BAPI PO accounts</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOACCOUNT_T</td>
<td>BAPI PO account</td>
</tr>
<tr>
<td>Table Type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>--------------------------------------------------------------</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOACCPROFITSE_T</td>
<td>BAPI PO account profit segment</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOADDRDELIVER_T</td>
<td>BAPI PO Address Delivery</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOCOMPONENTX_T</td>
<td>Update information for Components in BUS2012 API</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOCOMPONENT_T</td>
<td>BAPI structure for Components</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOCONDHEADERX_T</td>
<td>BAPI PO condition headers</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOCONDHEADER_T</td>
<td>BAPI PO condition header</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOCONDX_T</td>
<td>BAPI PO conditions</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOCOND_T</td>
<td>BAPI PO conditions</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOITEMX_T</td>
<td>BAPI PO Items table type</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOITEM_T</td>
<td>BAPI PO Item</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOSCHEDULEX_T</td>
<td>Delivery schedule (Change Parameter)</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOSCHEDULE_T</td>
<td>BAPI PO schedule</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOSCHEDULEUX_T</td>
<td>BAPI PO schedule</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOSHIPPEXP_T</td>
<td>Export structure for shipping data</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOTEXTHEADER_T</td>
<td>Header texts</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIMEPOTEXT_T</td>
<td>Item texts</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIPAREX_IN_T</td>
<td>Customer’s Own fields (Import Parameters)</td>
</tr>
<tr>
<td>/ARBA/IT_BAPIPAREX_OUT_T</td>
<td>Customer’s Own fields (export Parameters)</td>
</tr>
<tr>
<td>/ARBA/IT_NFM_BAPIDOCITM_T</td>
<td>BAPI Communication Structure NF Document Item Data</td>
</tr>
<tr>
<td>/ARBA/NFM_BAPIDOCITM_T</td>
<td>BAPI Communication Structure NF Document Item Data</td>
</tr>
</tbody>
</table>

/ARBA/PO_IMPORT Structures

The following are the structures for the /ARBA/PO_IMPORT package:

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/APIMEPOACC_PROF_SEGMENT</td>
<td>Reservation Profitability Segment</td>
</tr>
<tr>
<td>/ARBA/BAPIEIKP Foreign Trade</td>
<td>Export/Import: Header Data</td>
</tr>
<tr>
<td>/ARBA/BAPIEIKPX Foreign Trade</td>
<td>Import/Export Header Data (Change Parameter)</td>
</tr>
<tr>
<td>/ARBA/BAPIEIPPO Foreign Trade</td>
<td>Export/Import: Item Data</td>
</tr>
<tr>
<td>/ARBA/BAPIEIPPOX Foreign Trade</td>
<td>Import/Export: Item Data (Change Parameter)</td>
</tr>
<tr>
<td>/ARBA/BAPIEKKOP</td>
<td>Transfer Structure for Partner Roles in BAPI_PO_CREATE</td>
</tr>
<tr>
<td>Structure</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>/ARBA/BAPIESKLC Create Comm. Structure</td>
<td>Acct Asgt Distr. for Service Line</td>
</tr>
<tr>
<td>/ARBA/BAPIESLLC Communication Structure</td>
<td>Create Service Line</td>
</tr>
<tr>
<td>/ARBA/BAPIESLLTX</td>
<td>BAPI Services Long Text</td>
</tr>
<tr>
<td>/ARBA/BAPIESUCC Communication Structure</td>
<td>Contract Limits</td>
</tr>
<tr>
<td>/ARBA/BAPIESUHC</td>
<td>Structure for Limit</td>
</tr>
<tr>
<td>/ARBA/BAPIITEMSHIP BAPI</td>
<td>Shipping Data for Stock Transport Orders</td>
</tr>
<tr>
<td>/ARBA/BAPIITEMSHIPX</td>
<td>BAPI Shipping Data Change Parameter</td>
</tr>
<tr>
<td>/ARBA/BAPIMEDCM</td>
<td>Version Management</td>
</tr>
<tr>
<td>/ARBA/BAPIMEDCM_ALLVERSIONS</td>
<td>Version Management - All Version Data</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOACCOUNT Transfer Structure</td>
<td>Bapi PO Create Account Assignment</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOACCOUNTX</td>
<td>Account Assignment Fields in Purchase Order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOADDRDELIVERY PO Item</td>
<td>Address Structure BAPIADDR1 for Inbound Delivery</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOADDRVENDOR</td>
<td>Transfer Structure: Address of Vendor</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCOMPONENT_BAPI</td>
<td>Structure for Components</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCOMPONENTTX</td>
<td>Update Information for Components in BUS2012 API</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCOND</td>
<td>Conditions in Purchase Order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCONDHEADER</td>
<td>Transfer Structure: Bapi PO Create Condition Header</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCONDHEADERX</td>
<td>Conditions (Header, Change Bar)</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCONDX</td>
<td>Change Toolbar: Conditions in Purchase Order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOHEADER</td>
<td>Transfer Structure: Create PO header</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOHEADERX</td>
<td>Purchase Order Header Data (Change Parameter)</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOITEM</td>
<td>Transfer Structure: Bapi PO Create Item</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOITEMX</td>
<td>Purchase Order Item Data (Change Parameter)</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSCHEDULE</td>
<td>Transfer Structure: Bapi PO Create Schedule</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSCHEDULEUX</td>
<td>Fields for Schedule Lines in Purchase Order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSCHEDULEX</td>
<td>(Change Toolbar)</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSHIPPEXP</td>
<td>Export Structure for Shipping Data</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOTEXT</td>
<td>Transfer Structure: PO Create Item Text</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOTEXTHEADER</td>
<td>Transfer Structure: Bapi PO Create Header Text</td>
</tr>
<tr>
<td>/ARBA/BAPIPAREX</td>
<td>Ref. structure for BAPI parameter ExtensionIn/ExtensionOut</td>
</tr>
<tr>
<td>/ARBA/ERROR_MESSAGE</td>
<td>Error Message Structure for PO Push</td>
</tr>
<tr>
<td>/ARBA/NFM_BAPIDOCITM /NFM/</td>
<td>BAPI Communication Structure NF Document Item Data</td>
</tr>
</tbody>
</table>
### /ARBA/PO_IMPORT

The following is the BAdI information for the /ARBA/PO_IMPORT package:

<table>
<thead>
<tr>
<th>BAdI Class (ABAP Objects)</th>
<th>BAdI Interface (ABAP Objects)</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_EX_PO_IMPORT</td>
<td>/ARBA/IF_EX_PO_IMPORT</td>
<td>/ARBA/PO_IMPORT</td>
<td>BADI for PO Import</td>
</tr>
</tbody>
</table>

### /ARBA/CHANGEORDER Package

The /ARBA/CHANGEORDER package contains Ariba development objects for change and cancel order integration.

### /ARBA/CHANGEORDER Function

The /ARBA/CHANGEORDER package has the following function:

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CHANGEORDER_RFC</td>
<td>Holds change/cancel &amp; PO Header status function modules</td>
</tr>
</tbody>
</table>

### /ARBA/CHANGEORDER Table Types

The following are the table type structures in the /ARBA/CHANGEORDER package:

<table>
<thead>
<tr>
<th>Table Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BAPIESKLCL_T</td>
<td>Table type for Acct Assgt Distr. for service line</td>
</tr>
<tr>
<td>/ARBA/BAPIESLLLC_T</td>
<td>Table type for communication structure: Create Service Line</td>
</tr>
<tr>
<td>Table Type</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>/ARBA/BAPIESLLTX_T</td>
<td>Table type for BAPI Services Long Text</td>
</tr>
<tr>
<td>/ARBA/BAPIESUCC_T</td>
<td>Table type for communication structure: Contract Limits</td>
</tr>
<tr>
<td>/ARBA/BAPIESUHC_T</td>
<td>Table type for communication structure: Limits</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOACCOUNTPROFITS_T</td>
<td>Table type for BAPI_PROFITABILITY_SEGMENT</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOACCOUNTX_T</td>
<td>Table type for Account Assignment Fields in Purchase Order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOACCOUNT_T</td>
<td>Table type for Account Assignment Fields for Purchase Order</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOADDRDELIVERY_T</td>
<td>Table type for Address Structure BAPIADDR1 for Inbound Delivery</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCONDHEADERX_T</td>
<td>Table type for conditions (Header, Change Bar)</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCONDHEADER_T</td>
<td>Table type for conditions</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCONDX_T</td>
<td>Table type for conditions in purchase orders</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOCOND_T</td>
<td>Table type for conditions in purchase orders</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOITEMX_T</td>
<td>Table type for purchase order item data (Change Toolbar)</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOITEM_T</td>
<td>Table type for purchase order item</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSCHEDULE_T</td>
<td>Table type for purchase order delivery schedule fields</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSCHEDULX_T</td>
<td>Table type for schedule lines in purchase order fields</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOTEXTHEADER_T</td>
<td>Table type for header text</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOTEXT_T</td>
<td>Table type for texts: (Header or Item)</td>
</tr>
<tr>
<td>/ARBA/BAPIPAREX_T</td>
<td>Table type for BAPI parameter ExtensionIn/ExtensionOut</td>
</tr>
<tr>
<td>/ARBA/POHEADER_INFO_T</td>
<td>Table type for PO Header Info for Change Orders</td>
</tr>
<tr>
<td>/ARBA/PO_ACCOUNT_INFO_T</td>
<td>Table type for PO Account Info</td>
</tr>
<tr>
<td>/ARBA/PO_DELETE_ACCOUNTS_T</td>
<td>Table type for PO Delete Accounts</td>
</tr>
<tr>
<td>/ARBA/PO_DELETE_ITEMS_T</td>
<td>Table type for PO Delete Items</td>
</tr>
<tr>
<td>/ARBA/PO_HEADER_STATUS_T</td>
<td>Table type for PO Header Status Structure</td>
</tr>
<tr>
<td>/ARBA/PO_ITEM_T</td>
<td>Table type for PO Items for Change Orders (Ensure that you do not modify this table type)</td>
</tr>
<tr>
<td>/ARBA/PO_ITMS_T</td>
<td>Table type for PO Items</td>
</tr>
<tr>
<td>/ARBA/STRING_T</td>
<td>Table type for String</td>
</tr>
</tbody>
</table>
/ARBA/CHANGEORDER Structures

The following are the structures for the /ARBA/CHANGEORDER package:

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/POHEADER_INFO</td>
<td>PO Header information for Change Orders (Ensure that you do not modify this structure)</td>
</tr>
<tr>
<td>/ARBA/PO_ACCOUNT_INFO</td>
<td>PO Accounting information for Change Orders (Ensure that you do not modify this structure)</td>
</tr>
<tr>
<td>/ARBA/PO_DELETE_ACCOUNTS</td>
<td>PO Deleted Accounts for Change Orders (Ensure that you do not modify this structure)</td>
</tr>
<tr>
<td>/ARBA/PO_DELETE_ITEMS</td>
<td>PO Deleted Items for Change Orders (Ensure that you do not modify this structure)</td>
</tr>
<tr>
<td>/ARBA/PO_HEADER_STATUS</td>
<td>Ariba Extension for Order Header Status Structure</td>
</tr>
<tr>
<td>/ARBA/PO_ITEMS</td>
<td>PO Items for Change Orders (Ensure that you do not modify this structure)</td>
</tr>
<tr>
<td>/ARBA/STRING</td>
<td>Structure for text data for PO Push for Ariba</td>
</tr>
</tbody>
</table>

/ARBA/CHANGEORDER Data Element

The following is the element are the in the /ARBA/CHANGEORDER package:

<table>
<thead>
<tr>
<th>Data Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/SERIAL_NUM</td>
<td>Serial number of account assignment</td>
</tr>
</tbody>
</table>

BAdI

The following is the BAdI information for the /ARBA/CHANGEORDER package:

<table>
<thead>
<tr>
<th>BAdI Class (ABAP Objects)</th>
<th>BAdI Interface (ABAP Objects)</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_EX_PO_CHANGEORDER</td>
<td>/ARBA/IF_EX_PO_CHANGEORDER</td>
<td>/ARBA/PO_CHANGEORDER</td>
<td>BAdI for change purchase order</td>
</tr>
</tbody>
</table>

/ARBA/GOODS_RECEIPTS_CREATE Package

The /ARBA/GOODS_RECEIPTS_CREATE package comprises the function groups listed in the following section.
/ARBA/GOODS_RECEIPTS_CREATE Function Groups

The /ARBA/GOODS_RECEIPTS_CREATE package contains the following function groups:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/GOODSRECEIPTS_CREATE</td>
<td>Ariba goods receipts function group</td>
</tr>
<tr>
<td>/ARBA/GR_TMG</td>
<td>Table maintenance function group for the receipt table</td>
</tr>
</tbody>
</table>

/ARBA/GOODS_RECEIPTS_CREATE Structures

The /ARBA/GOODS_RECEIPTS_CREATE package contains the following structures:

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/ERROR_MESSAGE_CORE</td>
<td>Error message structure for po_push</td>
</tr>
<tr>
<td>/ARBA/GOODS_RECEIPTS_ERROR</td>
<td>Error message structure for goods receipt push</td>
</tr>
<tr>
<td>/ARBA/GOODS_RECEIPTS_HEADER</td>
<td>Goods receipt header information</td>
</tr>
<tr>
<td>/ARBA/GOODS_RECEIPTS_ITEM</td>
<td>Material document item information for Ariba goods receipt (Ensure that you do not modify this table.)</td>
</tr>
</tbody>
</table>

/ARBA/GOODS_RECEIPTS_CREATE Tables

The /ARBA/GOODS_RECEIPTS_CREATE package contains the following tables:

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/GR</td>
<td>Goods receipt pushed to SAP from Ariba</td>
</tr>
<tr>
<td>/ARBA/GRFLDSMAP</td>
<td>Maps interface tables to screen fields (This is maintained by the customer.)</td>
</tr>
<tr>
<td>/ARBA/GRFLDS_MAP</td>
<td>Maps interface tables to screen fields (Ensure that you do not modify this table.)</td>
</tr>
</tbody>
</table>

/ARBA/GOODS_RECEIPTS_CREATE Table Types

The /ARBA/GOODS_RECEIPTS>Create package contains the following table types:

<table>
<thead>
<tr>
<th>Table Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BDCDATA_T</td>
<td>Table type for BDCdata</td>
</tr>
<tr>
<td>/ARBA/GOODS_RECEIPTS_ERROR_T</td>
<td>Table type for error messages</td>
</tr>
</tbody>
</table>
Table Type

<table>
<thead>
<tr>
<th>Table Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/GOODS_RECEIPTS_ITEM_T</td>
<td>Table type for goods receipts item</td>
</tr>
</tbody>
</table>

Data Element

The following is the element in the /ARBA/GOODS_RECEIPTS_CREATE package:

<table>
<thead>
<tr>
<th>Data Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/FILE</td>
<td>Line Number</td>
</tr>
<tr>
<td></td>
<td>File number of the receipt</td>
</tr>
</tbody>
</table>

BAdI

The following is the BAdI information for the /ARBA/GOODS_RECEIPTS_CREATE package:

<table>
<thead>
<tr>
<th>BAdI Class (ABAP Objects)</th>
<th>BAdI Interface (ABAP Objects)</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_EX_GR_CREATE</td>
<td>/ARBA/IF_EX_GR_CREATE</td>
<td>/ARBA/GR_CREATE</td>
<td>Add-in for goods receipts creation</td>
</tr>
</tbody>
</table>

/ARBA/GR_INTEGRATION Package

The /ARBA/GR_INTEGRATION package is applicable for the Ariba goods receipts push using the file channel integration.

ARBA/GR_INTEGRATION Program

The ARBA/GR_INTEGRATION package contains the following programs:

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CR_GR_CREATE</td>
<td>Ariba goods receipts push for file channel integration</td>
</tr>
<tr>
<td>/ARBA/CR_GR_CREATEF01</td>
<td>Include for receipt program</td>
</tr>
</tbody>
</table>
ARBA/GR_INTEGRATION Structures

The ARBA/GR_INTEGRATION package contains the following structures:

Table 27: /ARBA/GR_INTEGRATION Structures

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/GR_CSV_LINEITEMS</td>
<td>Ariba customer CSV goods receipt lineitems structure</td>
</tr>
<tr>
<td>/ARBA/GR_CSV_NUMBERPULL</td>
<td>Ariba goods receipt push structure for the PO Number Pull response</td>
</tr>
</tbody>
</table>

BAdI

The following is the BAdI information for the ARBA/GR_INTEGRATION package:

<table>
<thead>
<tr>
<th>BAdI Class (ABAP Objects)</th>
<th>BAdI Interface (ABAP Objects)</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_EX_CREATE_GR</td>
<td>/ARBA/IF_EX_CREATE_GR</td>
<td>/ARBA/CREATE_GR</td>
<td>Add-in for Goods Receipts</td>
</tr>
</tbody>
</table>

/ARBA/FILE_CHANNEL Package

The /ARBA/FILE_CHANNEL package is applicable for the Ariba file channel objects.

/ARBA/FILE_CHANNEL Programs

The /ARBA/FILE_CHANNEL package contains the following programs:

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CI_BAPI_PO_COMMON</td>
<td>Include for PO create, change, cancel and header status pull</td>
</tr>
<tr>
<td>/ARBA/CR_BAPI_PO_CANCEL</td>
<td>Ariba Cancel Purchase Order for File Channel Integration</td>
</tr>
<tr>
<td>/ARBA/CR_BAPI_PO_CHANGE</td>
<td>Ariba Purchase Order Change for File Channel Integration</td>
</tr>
<tr>
<td>/ARBA/CR_BAPI_PO_CREATE1</td>
<td>PO Push File Channel Import</td>
</tr>
<tr>
<td>/ARBA/CR_PO_HEADER_STATUS</td>
<td>Ariba Purchase Order header status pull for File Channel</td>
</tr>
</tbody>
</table>
## Data Element

The following is the element in the /ARBA/FILE_CHANNEL package:

<table>
<thead>
<tr>
<th>Data Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/DATE_TIME_STAMP</td>
<td>DateTime stamp</td>
</tr>
</tbody>
</table>

## /ARBA/FILE_CHANNEL Function Group

The /ARBA/FILE_CHANNEL package contains the following function group:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/CREATE/ARBA/FILE_TABL</td>
<td>Table maintenance generator function group</td>
</tr>
</tbody>
</table>

## /ARBA/FILE_CHANNEL Structures

The /ARBA/FILE_CHANNEL package contains the following structures:

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CANCELPO_CSV_NUMBERPULL</td>
<td>Ariba CSV structure for Cancel PO response message</td>
</tr>
<tr>
<td>/ARBA/CPO_PUSH_CSV_ACCTDATA</td>
<td>Ariba CSV structure for Change PO accounting data</td>
</tr>
<tr>
<td>/ARBA/CPO_PUSH_CSV_DELACCOUNTS</td>
<td>Ariba CPO deleted accounts for File channel</td>
</tr>
<tr>
<td>/ARBA/CPO_PUSH_CSV_DEL_LINES</td>
<td>Ariba CPO deleted line items CSV for File channel</td>
</tr>
<tr>
<td>/ARBA/CPO_PUSH_CSV_LINEITEMS</td>
<td>Ariba CSV structure for Change PO push line items</td>
</tr>
<tr>
<td>/ARBA/CPO_PUSH_CSV_PO_TEXT</td>
<td>Ariba CPO text data CSV for File channel</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_ACCDATA</td>
<td>Ariba custom CSV structure for PO Account Assignment</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_HEADER</td>
<td>Ariba customer CSV PO Header Structure for PO</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_LADDetail</td>
<td>Ariba customer CSV PO Lineitems add details Structure for PO</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_LINEITEMS</td>
<td>Ariba customer CSV PO Lineitems Structure for PO</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_LINETEXT</td>
<td>Ariba customer CSV PO LINE Structure for PO</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_LINETEXT_NEW</td>
<td>Ariba structure for PO Line item text for reading CSV file</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_NUMBERPULL</td>
<td>Ariba PO Push Structure for PO Number Pull response</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_SCHEDULES</td>
<td>Ariba customer CSV Structure for PO Schedule lines</td>
</tr>
</tbody>
</table>
/ARBA/FILE_CHANNEL Table

The /ARBA/FILE_CHANNEL package contains the following table:

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/POH_DAT_TM</td>
<td>Ariba On-demand · Remittance date time storage</td>
</tr>
</tbody>
</table>

/ARBA/FILE_CHANNEL Table Types

The /ARBA/FILE_CHANNEL package contains the following table types:

<table>
<thead>
<tr>
<th>Table Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BAPIMEPOACCOUNTF_T</td>
<td>Table type for PO File Channel Accounting</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOADDRVENDORF_T</td>
<td>Table type for PO File Channel Vendor Address</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOHEADERF_T</td>
<td>Table type for File Channel Header Table Type</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOITEMF_T</td>
<td>Table type for PO File Channel Item Table Type</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOSCHEDULEF_T</td>
<td>Table type for PO File Channel Schedule item Table Type</td>
</tr>
<tr>
<td>/ARBA/BAPIMEPOTEXTTHEADR_T</td>
<td>Table type for Header Text</td>
</tr>
<tr>
<td>/ARBA/ERRORS_MESSAGE_T</td>
<td>Table type for Error Message</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_ACCDATA_T</td>
<td>Table type for PO PUSH CSV ACCDATA</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_HEADER_T</td>
<td>Table type for POPUSH CSV HEADER</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_LADDetail_T</td>
<td>Table type for POPUSH CSV LADDetail</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_LINEITEMS_T</td>
<td>Table type for POPUSH CSV LINEITEMS</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_LINETEXT_T</td>
<td>Table type for POPUSH CSV LINETEXT</td>
</tr>
<tr>
<td>/ARBA/POPUSH_CSV_SCHEDULES_T</td>
<td>Table type for POPUSH CSV SCHEDULES</td>
</tr>
<tr>
<td>/ARBA/PO_ACCOUNTS_INFO_T</td>
<td>Table type for PO account information</td>
</tr>
<tr>
<td>/ARBA/PO_IATEM_T</td>
<td>Table type for PO Items</td>
</tr>
<tr>
<td>/ARBA/STRINGS_T</td>
<td>Table type for string</td>
</tr>
</tbody>
</table>
BAdle

The following is the BAdI information for the /ARBA/FILE_CHANNEL package:

<table>
<thead>
<tr>
<th>BAdI Class (ABAP Objects)</th>
<th>BAdI Interface (ABAP Objects)</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_EX_PO_FILE_CHANNEL</td>
<td>/ARBA/IF_EX_PO_FILE_CHANNEL</td>
<td>/ARBA/PO_FILE_CHANNEL</td>
<td>Badi Definition For PO File Channel</td>
</tr>
</tbody>
</table>

/ARBA/INVOICE_INTEGRATION Package

The /ARBA/INVOICE_INTEGRATION package is applicable for the Ariba Invoice Integration package.

/ARBA/INVOICE_INTEGRATION Function Group

The /ARBA/INVOICE_INTEGRATION package contains the following function group:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/INVOICE_CREATE</td>
<td>Invoice verification function group</td>
</tr>
</tbody>
</table>

/ARBA/INVOICE_INTEGRATION Tables

The /ARBA/INVOICE_INTEGRATION package contains the following tables:

<table>
<thead>
<tr>
<th>Tables</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BAPIRETURN</td>
<td>Ariba return structure for invoices</td>
</tr>
<tr>
<td>/ARBA/BAPI_INVOICE_CREATE</td>
<td>Ariba invoice account data structure</td>
</tr>
<tr>
<td>/ARBA/INVOICE_CREATE_GLACCOUNT</td>
<td>Ariba GL account structure for invoices</td>
</tr>
<tr>
<td>/ARBA/INVOICE_CREATE_HEADER</td>
<td>Ariba header structure for invoices</td>
</tr>
<tr>
<td>/ARBA/INVOICE_CREATE_ITEM</td>
<td>Ariba item structure for invoices</td>
</tr>
</tbody>
</table>
The `/ARBA/INVOICE_INTEGRATION` package contains the following table types:

<table>
<thead>
<tr>
<th>Table Types</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/ARBA/BAPI_INVNCR_CREATE_MAT_T</code></td>
<td>Table type for BAPI material</td>
</tr>
<tr>
<td><code>/ARBA/BAPI_INVNCR_CREATE_TAX_T</code></td>
<td>Table type for BAPI tax</td>
</tr>
<tr>
<td><code>/ARBA/BAPI_INVNCR_CREATE_ADRDATA_T</code></td>
<td>Table type for BAPI address data</td>
</tr>
<tr>
<td><code>/ARBA/BAPI_INVNCR_CREATE_HEADER_T</code></td>
<td>Table type for BAPI Invoice header</td>
</tr>
<tr>
<td><code>/ARBA/BAPI_INVNCR_CREATE_ACCT_T</code></td>
<td>Table type for BAPI invoice create account</td>
</tr>
<tr>
<td><code>/ARBA/BAPI_INVNCR_CREATE_GLAC_T</code></td>
<td>Table type for BAPI GL account</td>
</tr>
<tr>
<td><code>/ARBA/BAPI_INVNCR_CREATE_ITEM_T</code></td>
<td>Table type for BAPI item data</td>
</tr>
<tr>
<td><code>/ARBA/BAPI_INVNCR_CREATE_WTAX_T</code></td>
<td>Table type for BAPI withholding tax</td>
</tr>
<tr>
<td><code>/ARBA/BAPI_INVNCR_CREATE_VENSPLIT_T</code></td>
<td>Table type for vendor invoice vendor split</td>
</tr>
<tr>
<td><code>/ARBA/BAPI_INVNCR_CREATE_T</code></td>
<td>Table type for invoice create</td>
</tr>
<tr>
<td><code>/ARBA/INVOICE_CREATE_HEADER_T</code></td>
<td>Table type for invoice create header</td>
</tr>
<tr>
<td><code>/ARBA/ITEM_DATA_T</code></td>
<td>Table type for item data</td>
</tr>
<tr>
<td><code>/ARBA/BAPIACAP03_T</code></td>
<td>Table type for vendor line items</td>
</tr>
<tr>
<td><code>/ARBA/BAPIACCR01_T</code></td>
<td>Table type for line item currency fields</td>
</tr>
<tr>
<td><code>/ARBA/BAPIACCRPO_T</code></td>
<td>Table type for posting in FI/CO: commitments (currency fields)</td>
</tr>
<tr>
<td><code>/ARBA/BAPIACGL03_T</code></td>
<td>Table type for G/L account line items</td>
</tr>
<tr>
<td><code>/ARBA/BAPIACKECR_T</code></td>
<td>Table type for posting in accounting: CO-PA account assignment characteristics</td>
</tr>
<tr>
<td><code>/ARBA/BAPIACKEVA_T</code></td>
<td>Table type for posting in FI/CO: CO-PA account assignment value fields</td>
</tr>
<tr>
<td><code>/ARBA/BAPIACPO00_T</code></td>
<td>Table type for posting in accounting: purchase order (item)</td>
</tr>
<tr>
<td><code>/ARBA/BAPIACTX01_T</code></td>
<td>Table type for posting to FI accounting: tax (item)</td>
</tr>
<tr>
<td><code>/ARBA/BAPIEXTC_T</code></td>
<td>Table type for container for Customer Exit parameter</td>
</tr>
<tr>
<td><code>/ARBA/BAPIRETURN_T</code></td>
<td>Table type for Ariba return table type</td>
</tr>
<tr>
<td><code>/ARBA/INVOICE_CREATE_GLACNT_T</code></td>
<td>Table type for Ariba GL account structure for invoice</td>
</tr>
</tbody>
</table>
BAdI

The following is the BAdI information for the /ARBA/INVOICE_INTEGRATION package:

<table>
<thead>
<tr>
<th>BAdI Class (ABAP Objects)</th>
<th>BAdI Interface (ABAP Objects)</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_EX_INVOICE_CREATE</td>
<td>/ARBA/IF_EX_INVOICE_CREATE</td>
<td>/ARBA/INVOICE_CREATE</td>
<td>Add-in for Invoice Create</td>
</tr>
</tbody>
</table>

/ARBA/EXPENSE_INTEGRATION Package

The following /ARBA/EXPENSE_INTEGRATION package is applicable for the Ariba Expense Report Integration package:

Related Information

Integrating non-PO (FI) invoices [page 232]
Ariba SAP Objects [page 280]

Function Groups

The /ARBA/EXPENSE_INTEGRATION package contains the following function groups:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/ER_INVOICE</td>
<td>Ariba ER Invoice Create function group</td>
</tr>
<tr>
<td>/ARBA/ER_TMG</td>
<td>Table maintenance function group</td>
</tr>
</tbody>
</table>

/ARBA/EXPENSE_INTEGRATION Table

The /ARBA/EXPENSE_INTEGRATION package contains the following table:

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/ER_GL_DATA</td>
<td>Offset G/L entry information for the Expense report Integration</td>
</tr>
</tbody>
</table>
/ARBA/EXPENSE_INTEGRATION Structures

The /ARBA/EXPENSE_INTEGRATION package contains the following structures:

<table>
<thead>
<tr>
<th>Structure Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/ER_CSV_HEADER</td>
<td>Ariba structure for CSV ER header</td>
</tr>
<tr>
<td>/ARBA/ER_CSV_LINE</td>
<td>Ariba structure for CSV ER line</td>
</tr>
<tr>
<td>/ARBA/ER_CSV_TAX</td>
<td>Ariba structure for CSV ER tax</td>
</tr>
</tbody>
</table>

/ARBA/EXPENSE_INTEGRATION Table Types

The /ARBA/EXPENSE_INTEGRATION package contains the following table types:

<table>
<thead>
<tr>
<th>Table Types</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BAPI_ERINV_CREATE_ITEM_T</td>
<td>Table type for BAPI item data</td>
</tr>
<tr>
<td>/ARBA/BAPI_ERINV_CREATE_MAT_T</td>
<td>Table type for BAPI material</td>
</tr>
<tr>
<td>/ARBA/BAPI_ERINV_CREATE_TAX_T</td>
<td>Table type for BAPI tax</td>
</tr>
<tr>
<td>/ARBA/BAPI_ERINV_CREATE_WTAX_T</td>
<td>Table type for BAPI withholding tax</td>
</tr>
<tr>
<td>/ARBA/BAPI_ERINV_CR_VENSPILT_T</td>
<td>Table type for vendor invoice vendor split</td>
</tr>
<tr>
<td>/ARBA/BAPI_ERIN_CREATE_ACCT_T</td>
<td>Table type for BAPI invoice create account</td>
</tr>
<tr>
<td>/ARBA/BAPI_ERIN_CREATE_GLAC_T</td>
<td>Table type for BAPI GL account</td>
</tr>
<tr>
<td>/ARBA/ER_CSV_HEADER_T</td>
<td>Table type for ER CSV header</td>
</tr>
<tr>
<td>/ARBA/ER_CSV_LINE_T</td>
<td>Table type for the general ledger account</td>
</tr>
<tr>
<td>/ARBA/ER_CSV_TAX_T</td>
<td>Table type for tax data</td>
</tr>
</tbody>
</table>

**BAdI**

The following is the BAdI information for the /ARBA/EXPENSE_INTEGRATION package:

<table>
<thead>
<tr>
<th>BAdI Class (ABAP Objects)</th>
<th>BAdI Interface (ABAP Objects)</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_EX_EXPENSE_CREATE</td>
<td>/ARBA/IF_EX_EXPENSE_CREATE</td>
<td>/ARBA/EXPENSE_CREATE</td>
<td>Add-in for Expense Create</td>
</tr>
</tbody>
</table>
/ARBA/EXPENSE_REPORT Package

The following /ARBA/EXPENSE_REPORT package is applicable for the Ariba Expense Report Push Integration package:

/ARBA/EXPENSE_REPORT Programs

The /ARBA/EXPENSE_REPORT package contains the following programs:

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CR_ER_INVOICE_CREATE</td>
<td>Read the ER files and create invoices</td>
</tr>
<tr>
<td>/ARBA/CR_ER_ERRORHANDLING</td>
<td>Include program for Expense Report</td>
</tr>
<tr>
<td>/ARBA/CR_ER_READERF01</td>
<td>Include program for Expense Report</td>
</tr>
</tbody>
</table>

/ARBA/EXPENSE_REPORT Function Group

The /ARBA/EXPENSE_REPORT package contains the following function group:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/READ_ER_FILE</td>
<td>Function group for Expense CSV reading</td>
</tr>
</tbody>
</table>

/ARBA/EXPENSE_REPORT Table Types

The /ARBA/EXPENSE_REPORT package contains the following table types:

<table>
<thead>
<tr>
<th>Table Types</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BAPI_INCINV_CRTE_ADATA_T</td>
<td>Table type for address data</td>
</tr>
<tr>
<td>/ARBA/BAPI_INCINV_CRTE_GLACC_T</td>
<td>Table type for invoice GL account</td>
</tr>
</tbody>
</table>
BAAdI

The following is the BAdI information for the /ARBA/EXPENSE_REPORT package:

<table>
<thead>
<tr>
<th>BAdI Class (ABAP Objects)</th>
<th>BAdI Interface (ABAP Objects)</th>
<th>BAdI Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CL_EX_READ_ER</td>
<td>/ARBA/IF_EX_READ_ER</td>
<td>/ARBA/READ_ER</td>
<td>Add-in for Expense report</td>
</tr>
</tbody>
</table>

/ARBA/REMITTANCE_FILE Package

The /ARBA/REMITTANCE_FILE package using the File Channel comprises the function groups listed in the following section.

/ARBA/REMITTANCE_FILE Program

The /ARBA/REMITTANCE_FILE package contains the following program:

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CR_REMITTANCE_EXPORT</td>
<td>Read the ER files and create invoices</td>
</tr>
</tbody>
</table>

/ARBA/REMITTANCE_FILE Function Group

The /ARBA/REMITTANCE_FILE package contains the following function group:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/REM_TMG</td>
<td>Table maintenance generator package</td>
</tr>
</tbody>
</table>

/ARBA/REMITTANCE_FILE Table

The /ARBA/REMITTANCE_FILE package contains the following table:

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/REM_DATETM</td>
<td>Ariba on-demand - Remittance Date Time Storage</td>
</tr>
</tbody>
</table>
/ARBA/CONTRACT Package

The /ARBA/CONTRACT package comprises the function groups listed in the following section.

/ARBA/CONTRACT Program

The /ARBA/CONTRACT_EXPORT package contains the following program:

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CONTRACT_EXPORT</td>
<td>Export contract information to Ariba On-Demand application</td>
</tr>
</tbody>
</table>

/ARBA/CONTRACT Function Groups

The /ARBA/CONTRACT_EXPORT package contains the following function groups:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CONTRACT</td>
<td>Function group for Contract RFC's</td>
</tr>
<tr>
<td>/ARBA/LAST_CONTR</td>
<td>Table maintenance function group</td>
</tr>
<tr>
<td>/ARBA/P2P_VENDOR</td>
<td>Table maintenance function group</td>
</tr>
</tbody>
</table>

/ARBA/CONTRACT Tables

The /ARBA/CONTRACT_EXPORT package contains the following tables:

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/LAST_CONTR</td>
<td>Ariba last-pulled contract document</td>
</tr>
<tr>
<td>/ARBA/P2P_VENDOR</td>
<td>Ariba procurement vendor table</td>
</tr>
</tbody>
</table>

/ARBA/CONTRACT Table Types

The /ARBA/CONTRACT package contains the following table types:

<table>
<thead>
<tr>
<th>Table Types</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CONTRACTACCOUNTINGS_T</td>
<td>Table type for Ariba Contract Accountings</td>
</tr>
</tbody>
</table>
### Table Types

<table>
<thead>
<tr>
<th>Table Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CONTRACTHEADER_T</td>
<td>Table type for Ariba Contract Header Details</td>
</tr>
<tr>
<td>/ARBA/CONTRACTLINEITEMS_T</td>
<td>Table type for Ariba Contract Line Item Details</td>
</tr>
<tr>
<td>/ARBA/CONTRACTTERMPRICING_T</td>
<td>Table type for Ariba Contract Pricing</td>
</tr>
<tr>
<td>/ARBA/CONTRACTTIEREDPRICING_T</td>
<td>Table type for Ariba Contract Tiered Pricing</td>
</tr>
<tr>
<td>/ARBA/LAST_CONTR_T</td>
<td>Table type for /ARBA/LAST_CONTR</td>
</tr>
<tr>
<td>/ARBA/RET_T</td>
<td>Table type for BAPIRET2</td>
</tr>
</tbody>
</table>

### /ARBA/CONTRACT Structures

The /ARBA/CONTRACT package contains the following structures:

<table>
<thead>
<tr>
<th>Structure Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CONTRACTACCOUNTINGS</td>
<td>Ariba structure for Contract Accounting Details</td>
</tr>
<tr>
<td>/ARBA/CONTRACTHEADER</td>
<td>Ariba structure for Contract Header Export</td>
</tr>
<tr>
<td>/ARBA/CONTRACTLINEITEMS</td>
<td>Ariba structure for Contract Line Item Details</td>
</tr>
<tr>
<td>/ARBA/CONTRACTTERMPRICING</td>
<td>Ariba structure for Contract Term Pricing</td>
</tr>
<tr>
<td>/ARBA/CONTRACTTIEREDPRICING</td>
<td>Ariba structure for Contract Tiered Pricing</td>
</tr>
</tbody>
</table>

### Enhancement Spot

The following is the Enhancement Sport information for the /ARBA/CONTRACT package:

<table>
<thead>
<tr>
<th>Enhancement Spot</th>
<th>Interface (ABAP Objects)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CONTRACT_DETAIL</td>
<td>/ARBA/IF_EX_CONT_DETAIL</td>
<td>Enhancement spot to customize contract details</td>
</tr>
</tbody>
</table>

### /ARBA/BUDGET_MGMT Package

This package contains the Ariba budget check objects.
/ARBA/BUDGET_MGMT Tables

The following is the table for the /ARBA/BUDGET_MGMT package:

Table 29: /ARBA/BUDGET_MGMT Table

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/TAX_CHECK</td>
<td>Check tax applied or not at company code level</td>
</tr>
</tbody>
</table>

/ARBA/BUDGET_MGMT Table Types

The following are the table types for the /ARBA/BUDGET_MGMT:

<table>
<thead>
<tr>
<th>Proxy Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BAPICOBL_AAT_T</td>
<td>BAPICOBL_AAT data type</td>
</tr>
<tr>
<td>/ARBA/BAPICOBL_T</td>
<td>Poss. derived coding block</td>
</tr>
<tr>
<td>/ARBA/RESPONSE_IMPORT_ITEM_T</td>
<td>Response Derivation Item Structure</td>
</tr>
</tbody>
</table>

/ARBA/BUDGET_MGMT Structures

The following are the structures for the /ARBA/BUDGET_MGMT package:

Table 30: /ARBA/BUDGET_MGMT Structures

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BAPICOBL</td>
<td>Poss. derived coding block</td>
</tr>
<tr>
<td>/ARBA/BAPICOBL_AAT</td>
<td>BAPICOBL_AAT data type</td>
</tr>
<tr>
<td>/ARBA/BAPICOBL_CUST</td>
<td>Customer fields to validate in coding block</td>
</tr>
<tr>
<td>/ARBA/BAPIMEREQACCOUNT</td>
<td>Transfer structure for requisition - account assignment</td>
</tr>
<tr>
<td>/ARBA/BAPIRETURN1</td>
<td>Funds Management return table</td>
</tr>
<tr>
<td>/ARBA/RESPONSE_IMPORT_ITEM</td>
<td>Response derivation item structure</td>
</tr>
</tbody>
</table>

/ARBA/BUDGET_MGMT Data Elements

The following are the elements in the /ARBA/BUDGET_MGMT package:

<table>
<thead>
<tr>
<th>Data Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/REQ_NO</td>
<td>Ariba requisition number</td>
</tr>
</tbody>
</table>
### /ARBA/BUDGET_MGMT Function Groups

The /ARBA/BUDGET_MGMT package contains the following function groups:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BUDGET_MGMT</td>
<td>Function group for budget management</td>
</tr>
<tr>
<td>/ARBA/DERIVATION</td>
<td>Function group to hold fund derivations</td>
</tr>
<tr>
<td>/ARBA/TAX_CHECK</td>
<td>Extended table maintenance (generated)</td>
</tr>
</tbody>
</table>

### /ARBA/BUDGET_MGMT Function Modules

The /ARBA/BUDGET_MGMT package contains the following function modules:

**Table 31: /ARBA/BUDGET_MGMT Function Group**

<table>
<thead>
<tr>
<th>Function Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/ACC_PURCHASE_REQUI_CHECK</td>
<td>Real-time budget check function module</td>
</tr>
</tbody>
</table>

**Table 32: /ARBA/DERIVATION Function Group**

<table>
<thead>
<tr>
<th>Function Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/DERIVATION</td>
<td>Function module to derive FM/GM related data from standard accounting elements</td>
</tr>
</tbody>
</table>

**Table 33: /ARBA/TAX_CHECK Function Group**

<table>
<thead>
<tr>
<th>Function Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TABLEFRAME_/ARBA/TAX_CHECK</td>
<td>Upper-level extended table maintenance</td>
</tr>
<tr>
<td>TABLEPROC_/ARBA/TAX_CHECK</td>
<td>Lower-level extended table maintenance</td>
</tr>
</tbody>
</table>

### /ARBA/BUDGET_MGMT Enhancement Spot

The following is the enhancement spot information for the /ARBA/BUDGET_MGMT package:

<table>
<thead>
<tr>
<th>Enhancement Spot</th>
<th>BAdl Interface (ABAP Objects)</th>
<th>BAdl Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/DERIVATION</td>
<td>/ARBA/IF_DERIVATION</td>
<td>/ARBA/DERIVATION</td>
<td>Derivation</td>
</tr>
</tbody>
</table>
/ARBA/ADVPYMNT package

This package contains the Ariba advance payment objects.

/ARBA/ADVPYMNT table types [page 317]
/ARBA/ADVPYMNT structures [page 317]
/ARBA/ADVPYMNT data elements [page 318]
/ARBA/ADVPYMNT programs [page 318]
/ARBA/ADVPYMNT function groups [page 318]
/ARBA/ADVPYMNT enhancement spot [page 319]

Related Information

Integrating advance payments [page 230]
Ariba SAP Objects [page 280]

/ARBA/ADVPYMNT table types

The /ARBA/ADVPYMNT package contains the following table types:

<table>
<thead>
<tr>
<th>Proxy Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/APR_REMDETAIL_T</td>
<td>Table type of /ARBA/APR_REMDETAIL</td>
</tr>
<tr>
<td>/ARBA/APR_REMHEADER_T</td>
<td>Table type of /ARBA/APR_REMHEADER</td>
</tr>
<tr>
<td>/ARBA/APR_REMIT_T</td>
<td>Table type of /ARBA/APR_REMIT</td>
</tr>
</tbody>
</table>

/ARBA/ADVPYMNT structures

The /ARBA/ADVPYMNT package contains the following structures:

Table 34: /ARBA/ADVPYMNT Structures

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/APR_CANCEL_CSV</td>
<td>Ariba CSV structure for advance payment request cancel</td>
</tr>
<tr>
<td>/ARBA/APR_CSV_HEADER</td>
<td>Ariba CSV header structure for advance payment request</td>
</tr>
<tr>
<td>/ARBA/APR_CSV_LINEITEM</td>
<td>Ariba CSV line item structure for advance payment request</td>
</tr>
<tr>
<td>/ARBA/APR_CSV_REMDETAIL</td>
<td>Advance payment remittance CSV item details</td>
</tr>
</tbody>
</table>
### Structure

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/APR_CSV_REMHEADER</td>
<td>Advance payment remittance CSV header</td>
</tr>
<tr>
<td>/ARBA/APR_CSV_RESPONSE_ERROR</td>
<td>Ariba CSV error response structure for APR</td>
</tr>
<tr>
<td>/ARBA/APR_CSV_RESPONSE_ID</td>
<td>Ariba CSV ID response structure for advance payment request</td>
</tr>
<tr>
<td>/ARBA/APR_HDR</td>
<td>Ariba advance payment header</td>
</tr>
<tr>
<td>/ARBA/APR_REMDETAIL</td>
<td>Advance payment remittance item details</td>
</tr>
<tr>
<td>/ARBA/APR_REMHEADER</td>
<td>Advance payment remittance header</td>
</tr>
</tbody>
</table>

### /ARBA/ADVPYMNT data elements

The /ARBA/ADVPYMNT package contains the following data elements:

<table>
<thead>
<tr>
<th>Data Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/APR_ID</td>
<td>Advance payment ID</td>
</tr>
<tr>
<td>/ARBA/ERP_ID</td>
<td>Advance payment request ID</td>
</tr>
</tbody>
</table>

### /ARBA/ADVPYMNT programs

The /ARBA/ADVPYMNT package contains the following programs:

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/CR_ADV_PAYMENT_CREATE</td>
<td>Export advance payment request from Ariba Procurement Solution to SAP ERP</td>
</tr>
<tr>
<td>/ARBA/CR_ADV_PAYMENTCANCEL</td>
<td>Export advance payment cancel request from Ariba Procurement Solution to SAP ERP</td>
</tr>
<tr>
<td>/ARBA/CR_ADV_PAY_REMIT_EXPORT</td>
<td>Export advance payment remittance from SAP ERP to Ariba Procurement Solution</td>
</tr>
</tbody>
</table>

### /ARBA/ADVPYMNT function groups

The /ARBA/ADVPYMNT package contains the following function groups:

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/ADV_PAYMENT_POST</td>
<td>Function module to create advance payment</td>
</tr>
<tr>
<td>/ARBA/ADV_PAYMENT_REV_POST</td>
<td>Function module to cancel advance payment</td>
</tr>
<tr>
<td>/ARBA/ADV_PAY_REMIT_EXPORT</td>
<td>Function module to export advance payment remittance</td>
</tr>
</tbody>
</table>
>**Function Group**

<table>
<thead>
<tr>
<th>Function Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/BTE_ADV_PAY_REMITTANCE</td>
<td>Function module for BTE advance payment remittance</td>
</tr>
</tbody>
</table>

**/ARBA/ADVYPYMNT enhancement spot**

The /ARBA/ADVYPYMNT package contains the following enhancement spot:

<table>
<thead>
<tr>
<th>Enhancement Spot</th>
<th>BAdl Interface (ABAP Objects)</th>
<th>BAdl Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/ARBA/ADV_PYMNT</td>
<td>/ARBA/IF_ADV_PYMNT</td>
<td>/ARBA/ADV_PYMNT</td>
<td>Advance payment</td>
</tr>
</tbody>
</table>
About SAP ERP Transports

If you are integrating your procurement solution with SAP ERP, you must install the SAP ERP transports as part of your Ariba Procurement Solution installation.

Ariba recommends that you ask your SAP BASIS personnel to install the Ariba-delivered transports.

Installing Transport Files [page 320]

Installing Transport Files

The process for installing transports is the same for both first-time installations and upgrades. Before you start, note the following:

- The transports contain both client-dependent and independent configurations. Therefore, specify a target client when adding these transports to the import queue. If you are using SAP’s Transport Management System (STMS), use extended transport control to specify the target client.
- Do not use umodes when installing the transports.
  In general, you do not use any umodes when installing these transports. Use umodes in the following instances:
    - You have a previous version of the Ariba objects in your SAP ERP, and some core objects have been modified. In this situation, override the repair flag on import by using a umode. Reinstall any overwritten modifications to Ariba core objects using the appropriate SAP ERP tools.
    - You need to re-import one of the delivered transports. In this case, make sure the transport is fully re-imported by using umode 1.
- Ariba recommends that you ask your SAP BASIS personnel to install the Ariba transports.
- The order in which you need to install the transports is specified in the Readme.txt file. This file is available in the ZIP file that you download from connect.ariba.com.
- If you do not require real-time integration, you do not have to import the real-time integration transport. However, if you do install this transport, configure your SAP ERP to use real-time integration with Ariba Procurement Solution after the import is complete.
About mapping workbooks

The mapping workbooks for Ariba Procurement Solution integrated with SAP ERP are intended as sample only. The buying organizations can customize mappings to suit their business needs.

- In case of outbound transactions, the data is transformed from XML to cXML and then uploaded to Ariba Procurement Solution.
- In case of inbound transactions, the data is transformed from cXML to XML and then transferred to SAP ERP.

The information about how the fields have been mapped between XML and cXML files for both inbound and outbound transactions is available in mapping workbooks provided by Ariba. You can download these mapping workbooks from connect.ariba.com. You as a buying organization can change the mapping of fields between XML and cXML according to your business requirements. For example, each buying organization lists requisitioners in cXML in a different way. These differences can be handled by customizing the mappings.

How to download the mapping workbooks

Procedure

1. Log in to connect.ariba.com using your user ID and password. If you do not have a User ID and Password for Ariba Connect, contact your Ariba account executive.
3. Under the Integration Tools section, click Integration tools for Ariba Procure-to-Pay. The Integration tools for Ariba Procure-to-Pay page appears.
5. Click Download and specify a location on your hard drive to download the mapping workbooks.

Results

The mapping workbooks for all transactions are available in a ZIP file.
Revision History

The following table provides a brief history of the updates to this guide. Ariba updates the technical documentation for its On Demand solutions if

- software changes delivered in service packs or hot fixes require a documentation update to correctly reflect the new or changed functionality;
- the existing content is incorrect or user feedback indicated that important content is missing.

Ariba reserves the right to update its technical documentation without prior notification. Most documentation updates will be made available in the same week as the software service packs are released, but critical documentation updates may be released at any time.

<table>
<thead>
<tr>
<th>Document Version</th>
<th>Month/Year of Update</th>
<th>Updated Chapter/Section</th>
<th>Short Description of Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>December 2016</td>
<td>N/A</td>
<td>Reset revision history for release Cloud Integration Release 9.0.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Multiple Chapters</td>
<td>Ariba Cloud Integration Release 9.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Updated throughout for the Ariba Cloud Integration Release 9.0.</td>
</tr>
<tr>
<td>2</td>
<td>January 2017</td>
<td>Ensuring that shipping and special handling are included as charges</td>
<td>Added a prerequisite.</td>
</tr>
</tbody>
</table>
Important Disclaimers and Legal Information

Coding Samples

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended to better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, unless damages were caused by SAP intentionally or by SAP's gross negligence.

Accessibility

The information contained in the SAP documentation represents SAP’s current view of accessibility criteria as of the date of publication; it is in no way intended to be a binding guideline on how to ensure accessibility of software products. SAP in particular disclaims any liability in relation to this document. This disclaimer, however, does not apply in cases of willful misconduct or gross negligence of SAP. Furthermore, this document does not result in any direct or indirect contractual obligations of SAP.

Gender-Neutral Language

As far as possible, SAP documentation is gender neutral. Depending on the context, the reader is addressed directly with "you", or a gender-neutral noun (such as "sales person" or "working days") is used. If when referring to members of both sexes, however, the third-person singular cannot be avoided or a gender-neutral noun does not exist, SAP reserves the right to use the masculine form of the noun and pronoun. This is to ensure that the documentation remains comprehensible.

Internet Hyperlinks

The SAP documentation may contain hyperlinks to the Internet. These hyperlinks are intended to serve as a hint about where to find related information. SAP does not warrant the availability and correctness of this related information or the ability of this information to serve a particular purpose. SAP shall not be liable for any damages caused by the use of related information unless damages have been caused by SAP's gross negligence or willful misconduct. All links are categorized for transparency (see: http://help.sap.com/disclaimer).